16-bit Proprietary Microcontroller

CMOS

F²MC-16F MB90246A Series

MB90246A

■ DESCRIPTION

The MB90246A series is a 16-bit microcontroller optimum to control mechatronics such as a hard disk drive unit.

The instruction set of F²MC-16F CPU core inherits AT architecture of F²MC*-16/16H family with additional instruction sets for high-level languages, extended addressing mode, enhanced multiplication/division instructions, and enhanced bit manipulation instructions. The microcontroller has a 32-bit accumulator for processing long word data (32-bit).

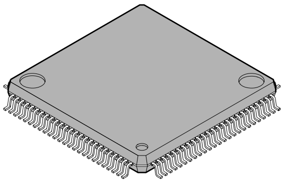
The MB90246A series contains a production addition unit as peripheral resources for enabling easy implementation of functions supported by IIR and FIR digital filters. It also supports a wealth of peripheral functions including:

- an 8/10-bit A/D converter having eight channels;
- an 8-bit D/A converter having three channels;
- UART;
- an 8-bit PWM timer having four channels;
- a timer having three plus one channels;
- an input capture (ICU) having two channels; and
- a DTP/external interrupt circuit having four channels.

* : F²MC stands for FUJITSU Flexible Microcontroller.

■ PACKAGE

100-pin Plastic LQFP



(FPT-100P-M05)

■ FEATURES

- Clock
  Operating clock can be selected from divided-by-2, 4, 8 or 32 of oscillation (at oscillation of 32 MHz, 1 MHz to 16 MHz).
  Minimum instruction execution time of 62.5 ns (at machine clock of 16 MHz)
- CPU addressing space of 16 Mbytes
  Internal addressing of 24-bit
  External accessing can be performed by selecting 8/16-bit bus width (external bus mode)
- Instruction set optimized for controller applications
  Rich data types (bit, byte, word, long word)
  Rich addressing mode (23 types)
  High code efficiency
  Enhanced precision calculation realized by the 32-bit accumulator
  Signed multiplication/division instruction
- Instruction set designed for high level language (C) and multi-task operations
  Adoption of system stack pointer
  Enhanced pointer indirect instructions
  Barrel shift instructions
- Enhanced execution speed
  8-byte instruction queue
- Enhanced interrupt function
  Priority levels: 8 levels
  External interrupt input ports: 4 ports
- Automatic data transmission function independent of CPU operation
  Extended intelligent I/O service function (EI^{2}OS)
- Low-power consumption (stand-by) mode
  Sleep mode (mode in which CPU operating clock is stopped)
  Stop mode (mode in which oscillation is stopped)
  Hardware stand-by mode
  Gear function
- Process
  CMOS technology
- I/O port
  General-purpose I/O ports (CMOS): 38
  General-purpose I/O ports (TTL): 11
  General-purpose I/O ports (N-ch open-drain): 8
  Total: 57
- Timer
  Timebase timer/watchdog timer: 1 channel
  8-bit PWM timer: 4 channels
  16-bit re-load timer: 3 channels
- 16-bit I/O timer
  16-bit free-run timer: 1 channel
  Input capture (ICU): 2 channels
- I/O simple serial interface
  Clock synchronized transmission can be used.
- UART: 1 channel
  Clock asynchronized or clock synchronized serial transmission can be selectively used.
- DTP/external interrupt circuit: 4 channels
  A module for starting extended intelligent I/O service (EI^{2}OS) and generating an external interrupt triggered by an external input.

(Continued)

(Continued)

- Delayed interrupt generation module
  Generates an interrupt request for switching tasks.
- 8/10-bit A/D converter: 8 channels
  8-bit or 10-bit resolution can be selectively used.
  Starting by an external trigger input.
- 8-bit D/A converter
  Resolution: 8 bits × 3 channels
- DSP interface for the IIR filter
  Function dedicated to IIR calculation
  Up to eight items of results of signed multiplication of 16 × 16 bits are added.
  Execution time of $Yk = \sum_{n=0}^{N} bn\ Yk - n + \sum_{m=0}^{M} am\ Xk - m$: 0.625 µs (When oscillation is 32 MHz and when N = M =3)
  Up to three N and M values can be set at your disposal.

■ PRODUCT LINEUP

Item \ Part number		MB90246A	MB90V246
Classification		Mass-produced product	Evaluation product
ROM size		None	
RAM size		4 k × 8 bits	6 k × 8 bits
CPU functions		The number of instructions: 412 Instruction bit length: 8 bits, 16 bits Instruction length: 1 byte to 7 bytes Data bit length: 1 bit, 4 bits, 8 bits, 16 bits, 32 bits Minimum execution time: 62.5 ns (at machine clock of 16 MHz) Interrupt processing time: 1.0 µs (at machine clock of 16 MHz, minimum value)	
Ports		General-purpose I/O ports (CMOS output): 38 General-purpose I/O ports (TTL input): 11 General-purpose I/O ports (N-ch open-drain output): 8 Total: 57	
Timebase timer		18-bit counter Interrupt interval: 0.256 ms, 1.024 ms, 4.096 ms, 16.384 ms (at oscillation of 32 MHz)	
Watchdog timer		Reset generation interval: 3.58 ms, 14.33 ms, 28.67 ms, 57.34 ms (at oscillation of 32 MHz, minimum value)	
8/16-bit PWM timer		Number of channels: 4 Pulse interval: 0.25 µs to 32.77 ms (at oscillation of 32 MHz)	
16-bit re-load timer		Number of channels: 3 16-bit re-load timer operation Interval: 125 ns to 131 ms (at machine clock of 16 MHz) External event count can be performed.	
16-bit I/O timer	16-bit free-run timer	Number of channel: 1 Overflow interrupts or intermediate bit interrupts may be generated.	
	Input capture (ICU)	Number of channel: 2 Rewriting a register value upon a pin input (rising, falling, or both edges)	
I/O simple serial interface		Number of channels: 2 Clock synchronized transmission (62.5 kbps to 8 Mbps)	
UART		Clock asynchronized transmission (2404 bps to 500 kbps) Clock synchronized transmission (250 kbps to 2 Mbps) Transmission can be performed by bi-directional serial transmission or by master/slave connection.	
DTP/external interrupt circuit		Number of inputs: 4 Started by a rising edge, a falling edge, an “H” level input, or an “L” level input. External interrupt circuit or extended intelligent I/O service (EI²OS) can be used.	
Delayed interrupt generation module		An interrupt generation module for switching tasks used in real-time operating systems.	

(Continued)

(Continued)

Item \ Part number	MB90246A	MB90V246
8/10-bit A/D converter	Conversion precision: 10-bit or 8-bit can be selectively used. Number of inputs: 8 One-shot conversion mode (converts selected channel only once) Continuous conversion mode (converts selected channel continuously) Stop conversion mode (converts selected channel and stop operation repeatedly)	
8-bit D/A converter	Number of channels: 3 Resolution: 8 bits Based on the R-2R system	
DSP interface for the IIR filter	Function dedicated to IIR calculation Up to 8 items of results of signed multiplication of 16 × 16 bits are added. Execution time of $Yk = \sum_{n=0}^{N} bn\ Yk - n + \sum_{m=0}^{M} am\ Xk - m$: 0.625 µs (When oscillation is 32 MHz and when N = M = 3) Up to three N and M values can be set at your disposal.	
Low-power consumption (stand-by) mode	Sleep/stop/hardware stand-by/gear function	
Process	CMOS	
Power supply voltage for operation*	4.5 V to 5.5 V	

* : Varies with conditions such as the operating frequency. (See section "■ Electrical Characteristics.") Assurance for the MB90V246 is given only for operation with a tool at a power voltage of 4.5 V to 5.5 V, an operating temperature of 0 to 70 degrees centigrade, and an clock frequency of 1.6 MHz to 32 MHz.

Note: A 64-word RAM for product addition is supported in addition to the above RAMs.

■ PACKAGE AND CORRESPONDING PRODUCTS

Package	MB90246A	MB90V246
FPT-100P-M05	○	×
PGA-256C-A02	×	○

○ : Available × : Not available

Note: For more information about each package, see section "■ Package Dimensions."

■ DIFFERENCES AMONG PRODUCTS

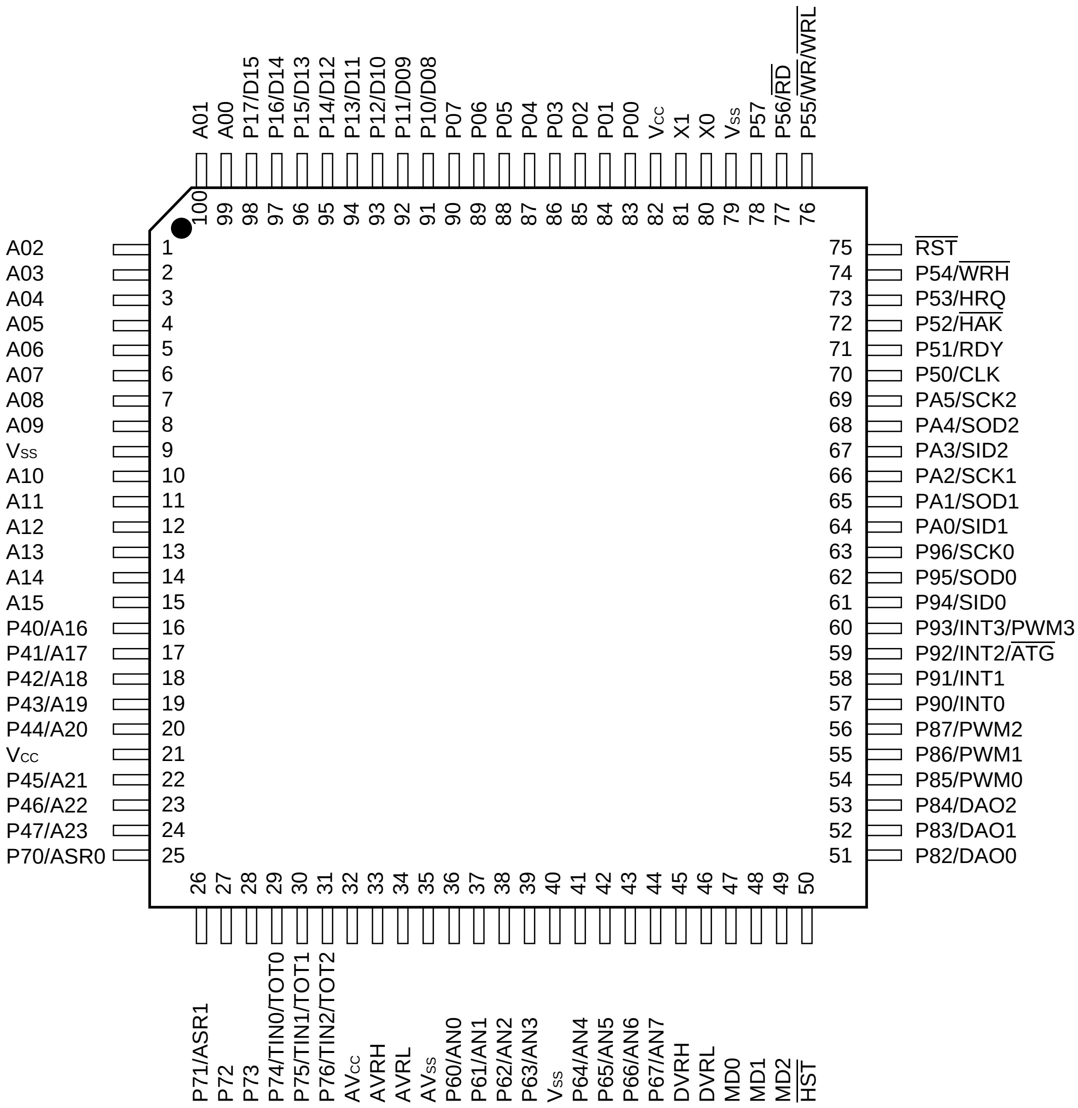
Memory Size

In evaluation with an evaluation chips, note the difference between the evaluation chip and the chip actually used.

The RAM size is 4 Kbytes for the MB90246A, and 6 Kbytes for the MB90V246.

■ PIN ASSIGNMENT

(Top view)



(FPT-100P-M05)

■ PIN DESCRIPTION

Pin no. LQFP*	Pin name	Circuit type	Function
80	X0	A	This is a crystal oscillator pin.
81	X1		
47 to 49	MD0 to MD2	C	This is an input pin for selecting operation modes. Connect directly to V_{CC} or V_{SS}.
75	$\overline{\text{RST}}$	B	This is external reset request signal.
50	$\overline{\text{HST}}$	C	This is a hardware stand-by input pin.
91 to 98	P10 to P17	D	This is a general-purpose I/O port. This function is valid in the 8-bit mode where the external bus is valid.
	D08 to D15		This is an I/O pin for the upper 8-bit of the external address data bus. This function is valid in the 16-bit mode where the external bus is valid.
16 to 20, 22 to 24	P40 to P44, P45 to P47	E	This is a general-purpose I/O port. This function becomes valid in the bit where the upper address control register is set to select a port.
	A16 to A20, A21 to A23		This is an output pin for the upper 8-bit of the external address bus. This function is valid in the mode where the external bus is valid and the upper address control register is set to select an address.
70	P50	E	This is a general-purpose I/O port. This function becomes valid when the CLK output is disabled.
	CLK		This is a CLK output pin. This function becomes valid when CLK output is enabled.
71	P51	D	This is a general-purpose I/O port. This function becomes valid when the external ready function are disabled.
	RDY		This is a ready input pin. This function becomes valid when the external ready function is enabled.
72	P52	D	This is a general-purpose I/O port. This function becomes valid when the hold function are disabled.
	$\overline{\text{HAK}}$		This is a hold acknowledge output pin. This function becomes valid when the hold function is enabled.
73	P53	D	This is a general-purpose I/O port. This function becomes valid when the hold function are disabled.
	HRQ		This is a hold request input pin. This function becomes valid when the hold function is enabled.
74	P54	E	This is a general-purpose I/O port. This function becomes valid, in the external bus 8-bit mode, or $\overline{\text{WRH}}$ pin output is disabled.
	$\overline{\text{WRH}}$		This is a write strobe output pin for the upper 8-bit of the data bus. This function becomes valid when the external bus 16-bit mode is selected, and $\overline{\text{WRH}}$ output pin is enabled.

* : FPT-100P-M05

(Continued)

Pin no. LQFP*	Pin name	Circuit type	Function
76	P55	E	This is a general-purpose I/O port. This function becomes valid when $\overline{WRL}$/$\overline{WR}$ pin output is disabled.
	$\overline{WR}$ $\overline{WRL}$		This is a write strobe output pin for the lower 8-bit of data bus. This function becomes valid when $\overline{WRL}$/$\overline{WR}$ pin output is enabled. $\overline{WRL}$ is used for holding the lower 8-bit for write strobe in 16-bit access operations, while $\overline{WR}$ is used for holding 8-bit data for write strobe in 8-bit access operations.
77	P56	E	This pin cannot be used as a general-purpose port.
	$\overline{RD}$		This is a read strobe output pin for the data bus. This function is valid in the mode where the external bus is valid.
78,28,27	P57,P73,P72	E	This is a general-purpose I/O port.
36 to 39, 41 to 44	P60 to P63, P64 to P67	G	This is an I/O port of an N-ch open-drain type. When the data register is read by a read instruction other than the modify write instruction with the corresponding bit in ADER set at “0”, the pin level is acquired. The value set in the data register is output to the pin as is.
	AN0 to AN3, AN4 to AN7		This is an analog input pin of the 8/10-bit A/D converter. When using this input pin, set the corresponding bit in ADER at “1”. Also, set the corresponding bit in the data register at “1”.
25	P70	E	This is a general-purpose I/O port.
	ASR0		This is a data input pin for input capture 0. Because this input is used as required when the input capture 0 is performing input operations, and it is necessary to stop outputs from other functions unless such outputs are made intentionally.
26	P71	E	This is a general-purpose I/O port.
	ASR1		This is a data input pin of input capture 1. Because this input is used as required when input capture 1 is performing input operations, and it is necessary to stop outputs by other functions unless such outputs are made intentionally.
29 to 31	P74 to P76	E	This is a general-purpose I/O port. This function becomes valid when outputs from 16-bit re-load timer 0 – 2 are disabled.
	TIN0 to TIN 2		This is an input pin of 16-bit timer. Because this input is used as required whin 16-bit timer 0 - 2 is performing input operations,and it is necessary to stop outputs by other functions unless such outputs are made intentionally.
	TOT0 to TOT2		These are output pins for 16-bit re-load timer 0 and 1. This function becomes valid when output from 16-bit re-load timer 0 – 2 are enabled.
51 to 53	P82 to P84	H	This is a general-purpose I/O port. This function becomes valid when data output from 8-bit D/A converter 0 – 2 are disabled.
	DAO0 to DAO2		This is an output pin of 8-bit D/A converter. This function becomes valid when data output from 8-bit D/A converter 0 – 2 are enabled.

* : FPT-100P-M05

(Continued)

Pin no. LQFP*	Pin name	Circuit type	Function
54 to 56	P85 to P87	E	This is a general-purpose I/O port. This function becomes valid when output from PWM0 – PWM2 are disabled.
	PWM0 to PWM2		This is an output pin of 8-bit PWM timer. This function becomes valid when output from PWM0 – PWM2 are enabled.
57, 58	P90, P91	F	This is a general-purpose I/O port.
	INT0, INT1		This is a request input pin of the DTP/external interrupt circuit ch.0 and 1. Because this input is used as required when the DTP/external interrupt circuit is performing input operations, and it is necessary to stop outputs from other functions unless such outputs are made intentionally.
59	P92	E	This is a general-purpose I/O port.
	INT2		This is an input pin of the DTP/external interrupt circuit ch.2. Because this input is used as required when the DTP/external interrupt circuit is performing input operations, and it is necessary to stop outputs from other functions unless such outputs are made intentionally.
	$\overline{\text{ATG}}$		This is a trigger input pin of the 8/10-bit A/D converter. Because this input is used as requited when the 8/10-bit A/D converter is performing input operations, and it is necessary to stop outputs by other functions unless such outputs are made intentionally.
60	P93	E	This is a general-purpose I/O port. This function is always valid. This function becomes valid when output from PWM3 is disabled.
	INT3		This is a request input of the DTP/external interrupt circuit ch. 3. Because this input is used as required when the DTP/external interrupt circuit is performing input operations, and it is necessary to stop outputs from other functions unless such output are made intentionally.
	PWM3		This is an output pin of 8-bit PWM timer. This function becomes valid when output from PWM3 is enabled.
61	P94	E	This is a general-purpose I/O port. This function becomes valid when serial data output from UART is disabled.
	SID0		This is a serial data I/O pin of UART. This function becomes valid when serial data output from UART is enabled. Because this input is used as required when UART is performing input operations, and it is necessary to stop outputs by other functions unless such outputs are made intentionally.

* : FPT-100P-M05

(Continued)

Pin no. LQFP*	Pin name	Circuit type	Function
62	P95	E	This is a general-purpose I/O port. This function becomes valid when data output from UART is disabled.
	SOD0		This is a data output pin of UART. This function becomes valid when data output from UART is enabled.
63	P96	E	This is a general-purpose I/O port. This function becomes valid when clock output from UART is disabled.
	SCK0		This is a clock I/O pin of UART. This function becomes valid when clock output from UART is enabled. Because this input is used as required when UART is performing input operations, and it is necessary to stop outputs by other functions unless such outputs are made intentionally.
1 to 6, 100, 99	A02 to A07, A01, A00	E	This is an output pin for the lower 8-bit of the external address bus.
7, 8, 10 to 15	A08, A09, A10 to A15	E	This is an output pin for the middle 8-bit of the external address bus. This function is valid in the mode where the external bus is valid and the middle address control refister is set to select an address.
64	PA0	E	This is a general-purpose I/O port.
	SID1		This is a data input pin of I/O simple serial interface 1. Because this input is used as required when I/O simple serial interface 1 is performing input operations, and it is necessarey to stop outputs by other functions unless such outputs are made intentionally.
65	PA1	E	This is a general-purpose I/O port. This function becomes valid when data output from I/O simple serial interface 1 is disabled.
	SOD1		This is a data output pin of I/O simple serial interface 1. This function becomes valid when data output from I/O simple serial interface 1 is enabled.
66	PA2	E	This is a general-purpose I/O port. This function becomes valid when clock output from I/O simple serial interface 1 is disabled.
	SCK1		This is a clock output pin of I/O simple serial interface 1. This function becomes valid when clock output from I/O simple serial interface 1 is enabled.

* : FPT-100P-M05

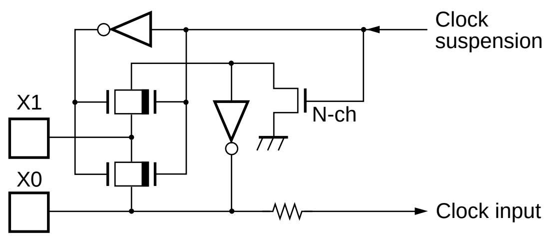
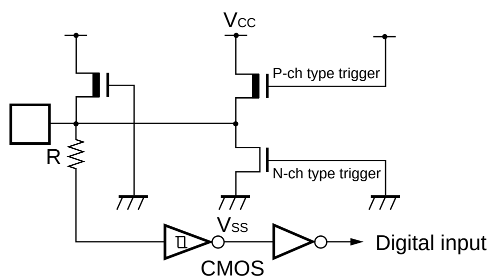
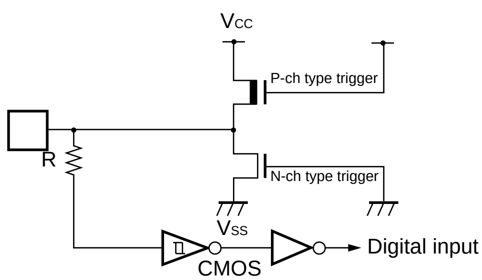
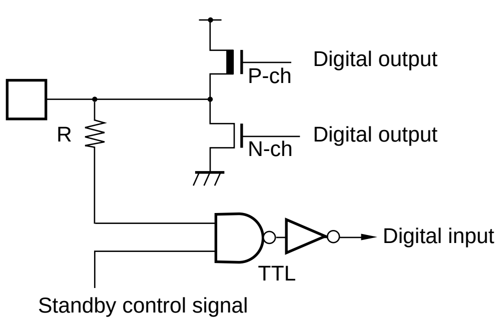
(Continued)

(Continued)

Pin no. LQFP*	Pin name	Circuit type	Function
67	PA3	E	This is a general-purpose I/O port.
	SID2		This is a data input pin of I/O simple serial interface 2. Because this input is used as required when is performing input operations, and it is I/O simple serial interface 2 necessarey to stop outputs by other functions unless such outputs are made intentionally.
68	PA4	E	This is a general-purpose I/O port. This function becomes valid when data output from I/O simple serial interface 2 is disabled.
	SOD2		This is a data output pin of I/O simple serial interface 2. This function becomes valid when data output from I/O simple serial interface 2 is enabled.
69	PA5	E	This is a general-purpose I/O port. This function becomes valid when clock output from I/O simple serial interface 2 is disabled.
	SCK2		This is clock output pin of I/O simple serial interface 2. This function becomes valid when clock output from I/O simple serial interface 2 is enabled.
83 to 90	D00 to D07	D	This is an I/O pin for the lower 8-bit of the external data bus.
21, 82	V_{CC}	Power supply	This is power supply to the digital circuit.
9, 40, 79	V_{SS}	Power supply	This is a ground level of the digital circuit.
32	AV_{CC}	Power supply	This is power supply to the analog circuit. Make sure to turn on/turn off this power supply with a voltage exceeding AV_{CC} applied to V_{CC}.
33	AVRH	Power supply	This is a reference voltage input to the A/D converter. Make sure to turn on/turn off this power supply with a voltage exceeding AVRH applied to AV_{CC}.
34	AVRL	Power supply	This is a reference voltage input to the A/D converter.
35	AV_{SS}	Power supply	This is a ground level of the analog circuit.
45	DVRH	Power supply	This is an external reference power supply pin for the D/A converter.
46	DVRL	Power supply	This is an external reference power supply pin for the D/A converter.

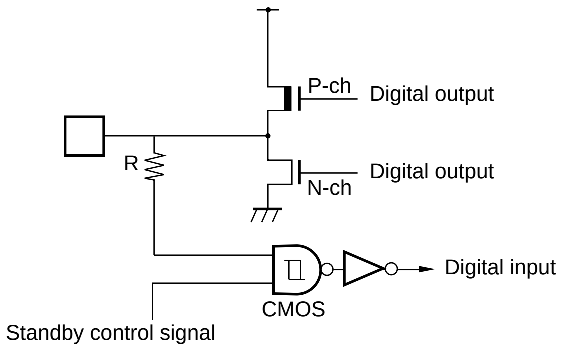
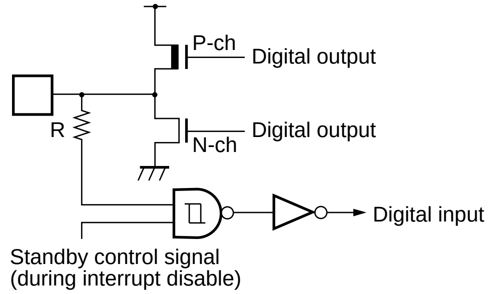
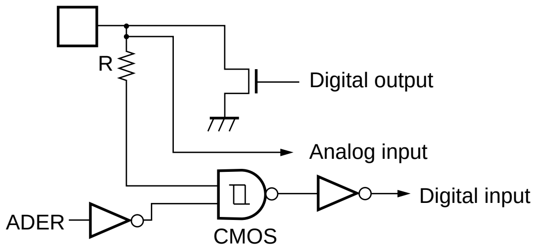
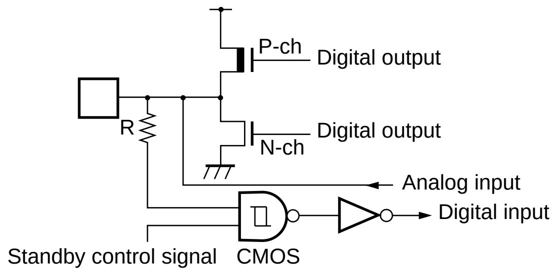
* : FPT-100P-M05

■ I/O CIRCUIT TYPE

Type	Circuit	Remarks
A		• For oscillation of 32 MHz • Oscillation feedback resistor approx. 1 MΩ
B		• CMOS level hysteresis input (without stand-by control) • Pull-up resistor approx. 50 kΩ
C		• CMOS level hysteresis input (without stand-by control)
D		• CMOS level output • TTL level input (with stand-by control)

(Continued)

(Continued)

Type	Circuit	Remarks
E		• CMOS level output • CMOS level hysteresis input (with stand-by control)
F		• CMOS level input • CMOS level hysteresis input (with stand-by control (during interrupt disable))
G		• N-ch open-drain • CMOS level output • CMOS level hysteresis input • Analog input (with analog control)
H		• CMOS level output • Analog output • CMOS level hysteresis input (with stand-by control)

■ HANDLING DEVICES

1. Make Sure that the Voltage not Exceed the Maximum Rating (to Avoid a Latch-up)

In CMOS ICs, a latch-up phenomenon is caused when an voltage exceeding V_{CC} or an voltage below V_{SS} is applied to input or output pins or a voltage exceeding the rating is applied across V_{CC} and V_{SS}.

When a latch-up is caused, the power supply current may be dramatically increased causing resultant thermal break-down of devices. To avoid the latch-up, make sure that the voltage not exceed the maximum rating.

In turning on/turning off the analog power supply, make sure the analog power voltage (AV_{CC}, AVRH) and analog input voltages not exceed the digital voltage (V_{CC}).

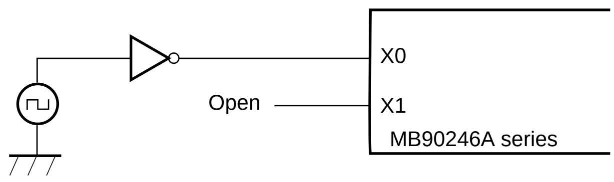
2. Connection of Unused Pins

Leaving unused pins open may result in abnormal operations. Clamp the pin level by connecting it to a pull-up or a pull-down resistor.

3. Notes on Using External Clock

In using the external clock, drive X0 pin only and leave X1 pin unconnected.

- **Using external clock**



4. Power Supply Pins

In products with multiple V_{CC} or V_{SS} pins, the pins of a same potential are internally connected in the device to avoid abnormal operations including latch-up. However, connect the pins external power and ground lines to lower the electro-magnetic emission level and abnormal operation of strobe signals caused by the rise in the ground level, and to conform to the total current rating.

Make sure to connect V_{CC} and V_{SS} pins via lowest impedance to power lines.

It is recommended to provide a bypass capacitor of around 0.1 µF between V_{CC} and V_{SS} pin near the device.

5. Crystal Oscillator Circuit

Noises around X0 or X1 pins may be possible causes of abnormal operations. Make sure to provide bypass capacitors via shortest distance from X0, X1 pins, crystal oscillator (or ceramic resonator) and ground lines, and make sure, to the utmost effort, that lines of oscillation circuit not cross the lines of other circuits.

It is highly recommended to provide a printed circuit board art work surrounding X0 and X1 pins with an grand area for stabilizing the operation.

6. Turning-on Sequence of Power Supply to A/D Converter, D/A Converter and Analog Inputs

Make sure to turn on the A/D converter power supply (AV_{CC}, AVRH, AVRL), D/A converter power supply and analog inputs (AN0 to AN7) after turning-on the digital power supply (V_{CC}).

Turn-off the digital power after turning off the A/D converter supply and analog inputs. In this case, make sure that the voltage not exceed AVRH or AV_{CC} (turning on/off the analog and digital supplies simultaneously is acceptable).

7. Connection of Unused Pins of A/D Converter

Connect unused pins of A/D converter to $AV_{CC} = V_{CC}$, AV_{SS} = AVRH = AVRL = V_{SS}.

8. “MOV @AL, AH”, “MOVW @AL, AH” Instructions

When the above instruction is performed to I/O space, an unnecessary writing operation may be performed (#FF, #FFFF) in the internal bus.
Use the compiler function for inserting an NOP instruction before the above instructions to avoid the writing operation.
Accessing RAM space with the above instruction does not cause any problem.

9. Initialization

In the device, there are internal registers which is initialized only by a power-on reset. To initialize these registers turning on the power again.

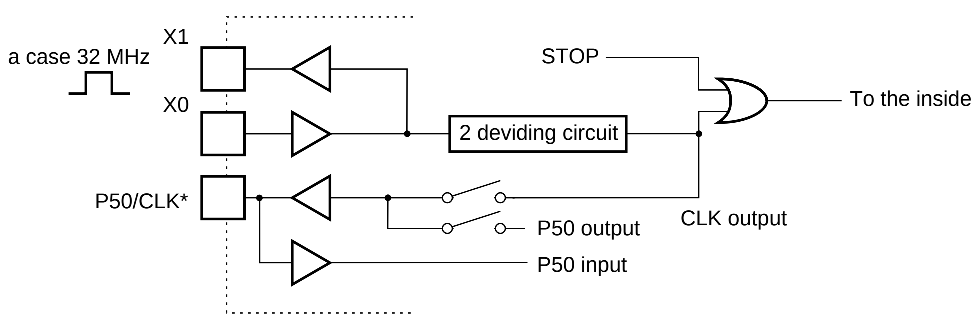
10.External Reset Input

To reset the internal securely, “L” level input to the $\overline{\text{RST}}$ pin must be at least 5 machine cycle.

11.$\overline{\text{HST}}$ Pin

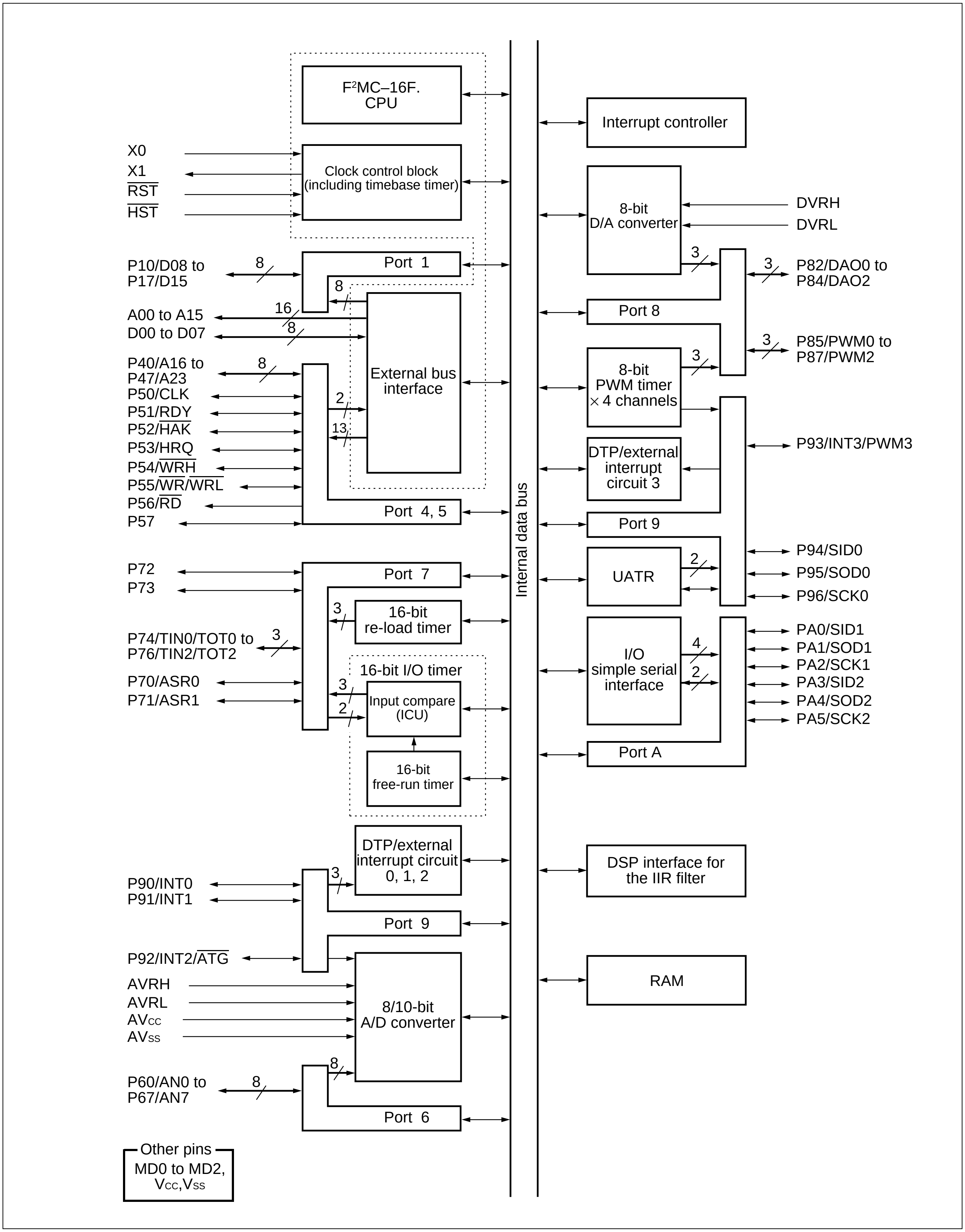
Make sure $\overline{\text{HST}}$ pin is set to “H” level when turn on the power supply. Also make sure $\overline{\text{HST}}$ pin is never set to “L” level, when $\overline{\text{RST}}$ pin is set to “L” level.

12.CLK Pin

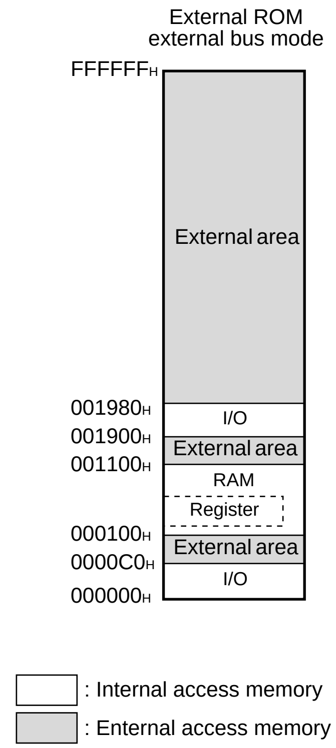


*: At P50/CLK pin in the external bus mode, CLK output is selected as an initial value.

■ BLOCK DIAGRAM



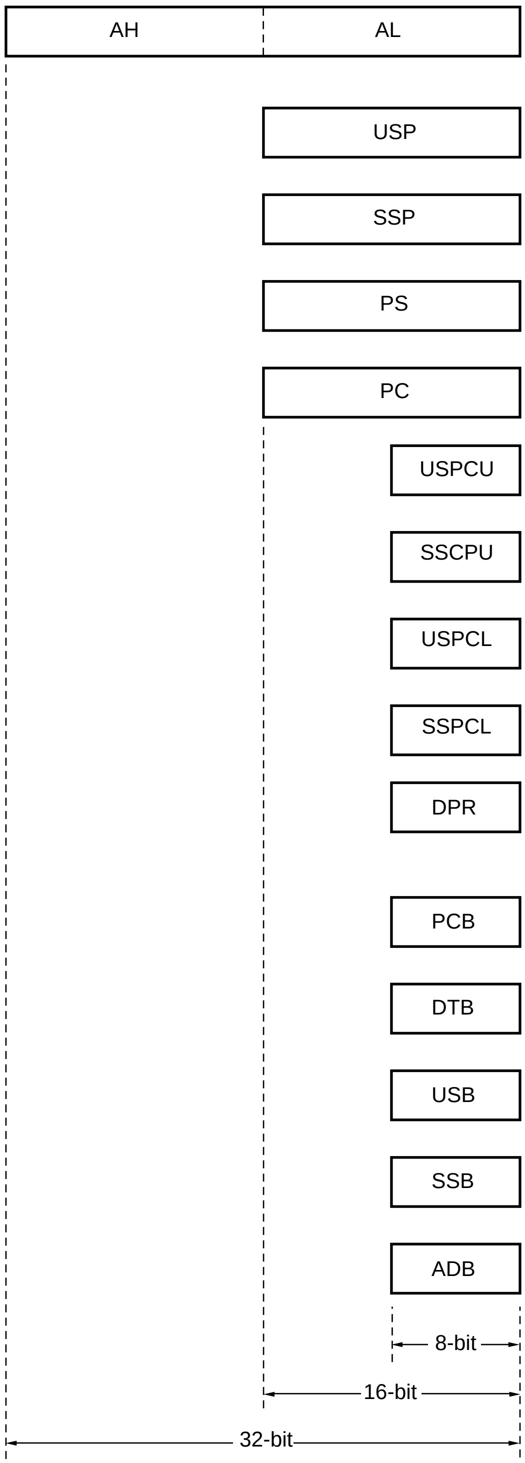
MEMORY MAP



The ROM data of bank FF is reflected in the upper address of bank 00, realizing effective use of the C compiler small model. The lower 16-bit of bank FF and the lower 16-bit of bank 00 is assigned to the same address, enabling reference of the table on the ROM without stating “far”.

■ F²MC-16F CPU PROGRAMMING MODEL

(1) Dedicated Registers



: Accumlator (A)
Dual 16-bit register used for storing results of calculation etc. The two 16-bit registers can be combined to be used as a 32-bit register.

: User stack pointer (USP)
The 16-bit pointer for containing a user stack address.

: System stack pointer (SSP)
The 16-bit pointer for displaying the status of the system stack address.

: Processor status (PS)
The 16-bit register for displaying the system status.

: Program counter (PC)
The 16-bit register for displaying storing location of the current instruction code.

: User stack upper limit register (USPCU)
The 16-bit register for specifying the upper limit of the user stack.

: System stack upper limit register (SSPCU)
The 16-bit register for specifying the upper limit of the system stack.

: User stack lower limit register (USPCL)
The 16-bit register for specifying the lower limit of the user stack.

: System stack lower limit register (SSPCL)
The 16-bit register for specifying the lower limit of the system stack.

: Direct page register (DPR)
The 8-bit register for specifying bit 8 through 15 of the operand address in the short direct addressing mode.

: Program bank register (PCB)
The 8-bit register for displaying the program space.

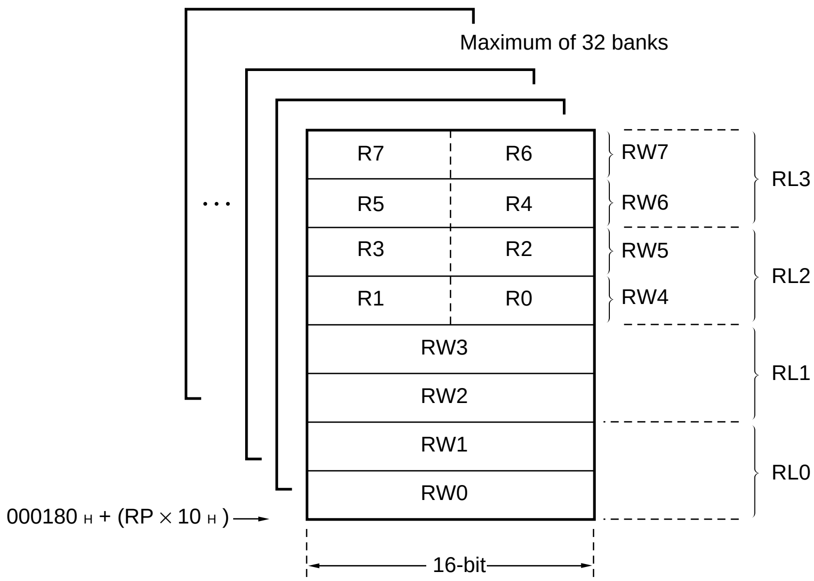
: Data bank register (DTB)
The 8-bit register for displaying the data space.

: User stack bank register (USB)
The 8-bit register for displaying the user stack space.

: System stack bank register (SSB)
The 8-bit register for displaying the system stack space.

: Additional data bank register (ADB)
The 8-bit register for displaying the additional data.

(2) General-purpose Registers



(3) Processor Status (PS)

	ILM			RP					CCR							
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
PS	ILM2	ILM1	ILM0	B4	B3	B2	B1	B0	—	I	S	T	N	Z	V	C
Initial value	0	0	0	0	0	0	0	0	—	0	1	X	X	X	X	X

— : Unused
X : Indeterminate

■ I/O MAP

Address	Abbreviated register name	Register name	Read/ write	Resource name	Initial value
000000H	(System reservation area)*1				
000001H	PDR1	Port 1 data register	R/W!	Port 1	XXXXXXXXB
000002H	(System reservation area)*1				
000003H					
000004H	PDR4	Port 4 data register	R/W!	Port 4	XXXXXXXXB
000005H	PDR5	Port 5 data register	R/W!	Port 5	XXXXXXXXB
000006H	PDR6	Port 6 data register	R/W!	Port 6	11111111B
000007H	PDR7	Port 7 data register	R/W!	Port 7	–XXXXXXXB
000008H	PDR8	Port 8 data register	R/W!	Port 8	XXXXXX––B
000009H	PDR9	Port 9 data register	R/W!	Port 9	–XXXXXXXB
00000AH	PDRA	Port A data register	R/W!	Port A	––XXXXXXB
00000BH to 00000FH	(Vacancy)				
000010H	(System reservation area)*1				
000011H	DDR1	Port 1 direction register	R/W	Port 1	00000000B
000012H	(System reservation area)*1				
000013H					
000014H	DDR4	Port 4 direction register	R/W	Port 4	00000000B
000015H	DDR5	Port 5 direction register	R/W	Port 5	00000000B
000016H	ADER	Analog input enable register	R/W	Port 6, 8/10-bit A/D converter	11111111B
000017H	DDR7	Port 7 direction register	R/W	Port 7	–0000000B
000018H	DDR8	Port 8 direction register	R/W	Port 8	000000––B
000019H	DDR9	Port 9 direction register	R/W	Port 9	–XXXXXXXB
00001AH	DDRA	Port A direction register	R/W	Port A	––000000B
00001BH to 00001FH	(Vacancy)				
000020H	SCR1	Serial control status register 1	R/W	I/O simple serial interface 1	10000000B
000021H	SSR1	Serial status register 1	R		–––––––1B
000022H	SDR1L	Serial data register 1 (L)	R/W		XXXXXXXXB
000023H	SDR1H	Serial data register 1 (H)	R/W		XXXXXXXXB

(Continued)

Address	Abbreviated register name	Register name	Read/ write	Resource name	Initial value
000024H	SCR2	Serial control status register 2	R/W	I/O simple serial interface 2	1 0 0 0 0 0 0 0 B
000025H	SSR2	Serial status register 2	R		– – – – – – – 1 B
000026H	SDR2L	Serial data register 2 (L)	R/W		X X X X X X X X B
000027H	SDR2H	Serial data register 2 (H)	R/W		X X X X X X X X B
000028H	UMC	Mode control register	R/W	UART	0 0 0 0 0 1 0 0 B
000029H	USR	Status register	R/W		0 0 0 1 0 0 0 0 B
00002AH	UIDR/ UODR	Input data register/ output data register	R/W		X X X X X X X X B
00002BH	URD	Rate and data register	R/W		0 0 0 0 0 0 0 0 B
00002CH	PWMC3	PWM3 operating mode control register	R/W	8-bit PWM timer 3	0 0 0 0 0 X X 1 B
00002DH	(Vacancy)				
00002EH	PRLL3	PWM3 re-road register (L)	R/W	8-bit PWM timer 3	X X X X X X X X B
00002FH	PRLH3	PWM3 re-road register (H)	R/W		X X X X X X X X B
000030H	ENIR	DTP/interrupt enable register	R/W	DTP/external interrupt circuit	– – – – 0 0 0 0 B
000031H	EIRR	DTP/interrupt factor register	R/W		– – – – 0 0 0 0 B
000032H	ELVR	Request level setting register	R/W		0 0 0 0 0 0 0 0 B
000033H	(Vacancy)				
000034H	PWMC0	PWM0 operating mode control register	R/W	8-bit PWM timer 0	0 0 0 0 0 X X 1 B
000035H	(Vacancy)				
000036H	PRLL0	PWM0 re-road register (L)	R/W	8-bit PWM timer 0	X X X X X X X X B
000037H	PRLH0	PWM0 re-road register (H)	R/W		X X X X X X X X B
000038H	PWMC1	PWM1 operating mode control register	R/W	8-bit PWM timer 1	0 0 0 0 0 X X 1 B
000039H	(Vacancy)				
00003AH	PRLL1	PWM1 re-road register (L)	R/W	8-bit PWM timer 1	X X X X X X X X B
00003BH	PRLH1	PWM1 re-road register (H)	R/W		X X X X X X X X B
00003CH	PWMC2	PWM2 operating mode control register	R/W	8-bit PWM timer 2	0 0 0 0 0 X X 1 B
00003DH	(Vacancy)				
00003EH	PRLL2	PWM2 re-road register (L)	R/W	8-bit PWM timer 2	X X X X X X X X B
00003FH	PRLH2	PWM2 re-road register (H)	R/W		X X X X X X X X B
000040H	TMCSR0	Timer control status register 0 lower digits	R/W	16-bit re-load timer 0	0 0 0 0 0 0 0 0 B
000041H		Timer control status register 0 upper digits	R/W		– – – – 0 0 0 0 B

(Continued)

Address	Abbreviated register name	Register name	Read/ write	Resource name	Initial value
000042H	TMR0	16-bit timer register 0	R	16-bit re-load timer 0	XXXXXXXXB
000043H					XXXXXXXXB
000044H	TMRLR0	16-bit re-load register 0	R/W		XXXXXXXXB
000045H					XXXXXXXXB
000046H	(Vacancy)				
000047H					
000048H	TMCSR1	Timer control status register 1 lower digits	R/W	16-bit re-load timer 1	00000000B
000049H		Timer control status register 1 upper digits	R/W		––––0000B
00004AH	TMR1	16-bit timer register 1	R		XXXXXXXXB
00004BH					XXXXXXXXB
00004CH	TMRLR1	16-bit re-load register 1	R/W		XXXXXXXXB
00004DH					XXXXXXXXB
00004EH	(Vacancy)				
00004FH					
000050H	TMCSR2	Timer control status register 2 lower digits	R/W	16-bit re-load timer 2	00000000B
000051H		Timer control status register 2 upper digits	R/W		––––1111B
000052H	TMR2	16-bit timer register 2	R		XXXXXXXXB
000053H					XXXXXXXXB
000054H	TMRLR2	16-bit re-load register 2	R/W		XXXXXXXXB
000055H					XXXXXXXXB
000056H to 000059H	(Vacancy)				
00005AH	DADR0	D/A data register 0	R/W	8-bit D/A converter 0	XXXXXXXXB
00005BH	DACR0	D/A control register 0	R/W		–––––––0B
00005CH	DADR1	D/A data register 1	R/W	8-bit D/A converter 1	XXXXXXXXB
00005DH	DACR1	D/A control register 1	R/W		–––––––0B
00005EH	DADR2	D/A data register 2	R/W	8-bit D/A converter 2	XXXXXXXXB
00005FH	DACR2	D/A control register 2	R/W		–––––––0B
000060H	IPCP0	Input capture register 0	R	16-bit I/O timer (input capture 0, 1)	XXXXXXXXB
000061H					XXXXXXXXB
000062H	IPCP1	Input capture register 1	R		XXXXXXXXB
000063H					XXXXXXXXB
000064H	ICS0	Input capture control register	R/W		00000000B

(Continued)

Address	Abbreviated register name	Register name	Read/ write	Resource name	Initial value
000065H to 00006BH	(Vacancy)				
00006CH	TCDT	Timer data register	R/W	16-bit I/O timer (16-bit free-run timer)	0 0 0 0 0 0 0 0 B
00006DH					0 0 0 0 0 0 0 0 B
00006EH	TCCS	Timer control status register	R/W		0 0 0 0 0 0 0 0 B
00006FH	(Vacancy)				
000070H	ADCSL	A/D control status register lower digits	R/W	8/10-bit A/D converter	0 0 0 – 0 0 0 0 B
000071H	ADCSH	A/D control status register upper digits	R/W		– 0 0 0 – – 0 0 B
000072H	ADCT	Conversion time setting register	R/W		X X X X X X X X B
000073H					X X X X X X X X B
000074H	ADTL0	A/D data register 0	R		X X X X X X X X B
000075H	ADTH0		R		– – – – – – * * B
000076H	ADTL1	A/D data register 1	R		X X X X X X X X B
000077H	ADTH1		R		– – – – – – * * B
000078H	ADTL2	A/D data register 2	R		X X X X X X X X B
000079H	ADTH2		R		– – – – – – * * B
00007AH	ADTL3	A/D data register 3	R		X X X X X X X X B
00007BH	ADTH3		R		– – – – – – * * B
00007CH to 00007FH	(Vacancy)				
000080H	MCSR	Product addition control status register lower digits	R/W	DSP interface for the IIR filter	X X X 0 X X X 0 B
000081H		Product addition control status register digits	R/W		– X X X X X X X B
000082H	MCCRL	Product addition continuation control register lower digits	R/W		0 0 0 0 0 0 0 0 B
000083H	MCCRH	Product addition continuation control register upper digits	R/W		– – – – – – 0 0 B
000084H	MDORL	Production addition output register	R		X X X X X X X X B
000085H					X X X X X X X X B
000086H	MDORM		R		X X X X X X X X B
000087H	MDORH		R		X X X X X X X X B
000088H					X X X X X X X X B

(Continued)

(Continued)

Address	Abbreviated register name	Register name	Read/ write	Resource name	Initial value
000089H to 00008FH	(Vacancy)				
000090H to 00009EH	(System reservation area)*1				
00009FH	DIRR	Delayed interrupt factor generation/ cancellation register	R/W	Delayed interrupt generation module	– – – – – – – 0 B
0000A0H	STBYC	Standby control register	R/W	Low-power consumption (stand-by) mode	0 0 0 1 X X X X B
0000A1H to 0000A3H	(System reservation area)*1				
0000A4H	HACR	Upper address control register	W	External bus pin	*2
0000A5H	EPCR	External pin control register	W		*2
0000A8H	WDTC	Watchdog timer control register	R/W	Watchdog timer	X X X X X X X X B
0000A9H	TBTC	Timebase timer control register	R/W	Timebase timer	– X X 0 0 1 0 0 B
0000B0H	ICR00	Interrupt control register 00	R/W	Interrupt controller	0 0 0 0 0 1 1 1 B
0000B1H	ICR01	Interrupt control register 01	R/W		0 0 0 0 0 1 1 1 B
0000B2H	ICR02	Interrupt control register 02	R/W		0 0 0 0 0 1 1 1 B
0000B3H	ICR03	Interrupt control register 03	R/W		0 0 0 0 0 1 1 1 B
0000B4H	ICR04	Interrupt control register 04	R/W		0 0 0 0 0 1 1 1 B
0000B5H	ICR05	Interrupt control register 05	R/W		0 0 0 0 0 1 1 1 B
0000B6H	ICR06	Interrupt control register 06	R/W		0 0 0 0 0 1 1 1 B
0000B7H	ICR07	Interrupt control register 07	R/W		0 0 0 0 0 1 1 1 B
0000B8H	ICR08	Interrupt control register 08	R/W		0 0 0 0 0 1 1 1 B
0000B9H	ICR09	Interrupt control register 09	R/W		0 0 0 0 0 1 1 1 B
0000BAH	ICR10	Interrupt control register 10	R/W		0 0 0 0 0 1 1 1 B
0000BBH	ICR11	Interrupt control register 11	R/W		0 0 0 0 0 1 1 1 B
0000BCH	ICR12	Interrupt control register 12	R/W		0 0 0 0 0 1 1 1 B
0000BDH	ICR13	Interrupt control register 13	R/W		0 0 0 0 0 1 1 1 B
0000BEH	ICR14	Interrupt control register 14	R/W		0 0 0 0 0 1 1 1 B
0000BFH	ICR15	Interrupt control register 15	R/W		0 0 0 0 0 1 1 1 B
0000C0H to 0000FFH	(External area)*3				

Descriptions for read/write
- R/W: Readable and writable
- R: Read only
- W: Write only
- R/W!: Bits for reading operation only or writing operation only are included. Refer to the register lists for specific resource for detailed information.

Descriptions for initial value
- 0 : The initial value of this bit is “0”.
- 1 : The initial value of this bit is “1”.
- X : The initial value of this bit is indeterminate.
- – : This bit is not used. The initial value is indeterminate.
- * : The storage type varies with the value of the ADCSH CREG bit.

*1: Access prohibited.
*2: The initial value varies with bus mode.
*3: This area is the only external access area having an address of $0000FF_H$ or lower. Access to any of the addresses specified as reserved areas in the table is handled as if an internal area were accessed. A signal for accessing an external bus is not generated.
*4: When a register described as R/W! or W in the read/write column is accessed by a bit setting instruction or other read modify write instructions, the bit pointed to by the instruction becomes a set value. If a bit is writable by other bits, however, malfunction occurs. You must not, therefore, access that register using these instructions.

Note: For bits that is initialized by an reset operation, the initial value set by the reset operation is listed as an initial value. Note that the values are different from reading results.

■ INTERRUPT FACTORS, INTERRUPT VECTORS, INTERRUPT CONTROL REGISTER

Interrupt source	EI^{2}OS support	Interrupt vector			Interrupt control register		Priority[*2]
		Number		Address	ICR	Address	
Reset	×	# 08	08$_H$	FFFFDC$_H$	—	—	High
INT9 instruction	×	# 09	09$_H$	FFFFD8$_H$	—	—	↑
Exception	×	# 10	0A$_H$	FFFFD4$_H$	—	—	
DTP/external interrupt circuit Channel 0	○	# 11	0B$_H$	FFFFD0$_H$	ICR00	0000B0$_H$	
DTP/external interrupt circuit Channel 1	○	# 13	0D$_H$	FFFFC8$_H$	ICR01	0000B1$_H$	
Input capture (ICU) Channel 0	○	# 15	0F$_H$	FFFFC0$_H$	ICR02	0000B2$_H$	
Input capture (ICU) Channel 1	△	# 17	11$_H$	FFFFB8$_H$	ICR03	0000B3$_H$	
I/O simple serial interface Channel 2	△	# 18	12$_H$	FFFFB4$_H$			
DTP/external interrupt circuit Channel 2	○	# 19	13$_H$	FFFFB0$_H$	ICR04	0000B4$_H$	
DTP/external interrupt circuit Channel 3	○	# 21	15$_H$	FFFFA8$_H$	ICR05	0000B5$_H$	
16-bit free-run timer Overflow	○	# 23	17$_H$	FFFFA0$_H$	ICR06	0000B6$_H$	
Timebase timer Interval interrupt	○	# 25	19$_H$	FFFF98$_H$	ICR07	0000B7$_H$	
16-bit re-load timer Channel 0	○	# 27	1B$_H$	FFFF90$_H$	ICR08[*1]	0000B8$_H$	
8-bit PWM timer Channel 0	×	# 28	1C$_H$	FFFF8C$_H$			
16-bit re-load timer Channel 1	○	# 29	1D$_H$	FFFF88$_H$	ICR09[*1]	0000B9$_H$	
8-bit PWM timer Channel 1	×	# 30	1E$_H$	FFFF84$_H$			
16-bit re-load timer Channel 2	○	# 31	1F$_H$	FFFF80$_H$	ICR10[*1]	0000BA$_H$	
8-bit PWM timer Channel 2	×	# 32	20$_H$	FFFF7C$_H$			
8/10-bit A/D converter measurement complete	○	# 33	21$_H$	FFFF78$_H$	ICR11[*1]	0000BB$_H$	
8-bit PWM timer Channel 3	×	# 34	22$_H$	FFFF74$_H$			
I/O simple serial interface Channel 1	○	# 35	23$_H$	FFFF70$_H$	ICR12	0000BC$_H$	
UART transmission complete	○	# 37	25$_H$	FFFF68$_H$	ICR13	0000BD$_H$	
UART reception complete	◎	# 39	27$_H$	FFFF60$_H$	ICR14	0000BE$_H$	
Delayed interrupt generation module	×	# 42	2A$_H$	FFFF54$_H$	ICR15	0000BF$_H$	↓
Stack fault	×	# 255	FF$_H$	FFFC00$_H$	—	—	Low

○ : Can be used

× : Can not be used

◎ : Can be used. With Extended intelligent I/O service (EI^{2}OS) stop function at abnormal operation.

△ : Can be used if interrupt request using ICR are not commonly used.

*1: • Interrupt levels for peripherals that commonly use the ICR register are in the same level.
- When the extended intelligent I/O service (EI^2OS) is specified in a peripheral device commonly using the ICR register, only one of the functions can be used.
- When the extended intelligent I/O service (EI^2OS) is specified for one of the peripheral functions, interrupts can not be used on the other function.

*2: The level shows priority of same level of interrupt invoked simultaneously.

■ PERIPHERALS

1. I/O Port

(1) Input/output Port

Ports 1, 4, 5, 7 to 9, A are general-purpose I/O ports having a combined function as an external bus pin and a resource input. The input output ports function as general-purpose I/O port only in the single-chip mode. In the external bus mode, the ports are configured as external bus pins, and part of pins for port 4 can be configured as general-purpose I/O port by setting the bus control signal select register (ECSR).

- Operation as output port

  The pin is configured as an output port by setting the corresponding bit of the DDR register to “1”.
  Writing data to PDR register when the port is configured as output, the data is retained in the output latch in the PDR and directly output to the pin.

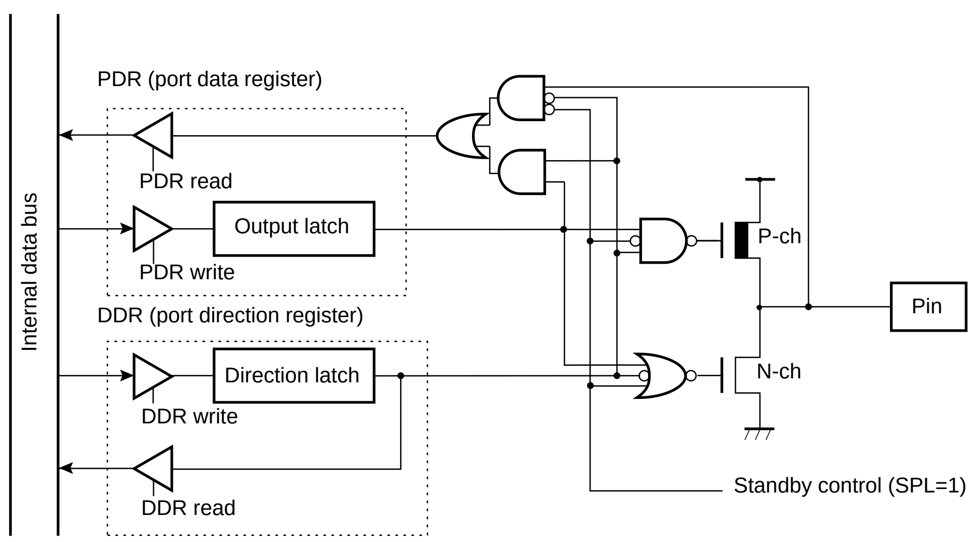
  The value of the pin (the same value retained in the output latch of PDR) can be read out by reading the PDR register.

Note: When a read-modify-write type instruction (e.g. bit set instruction) is performed to the port data register, the destination bit of the operation is set to the specified value, not affecting the bits configured by the DDR register for output, however, values of bits configured by the DDR register as inputs are changed because input values to the pins are written into the output latch. To avoid this situation, configure the pins by the DDR register as output after writing output data to the PDR register when configuring the bit used as input as outputs.

- Operation as input port

  The pin is configured as an input by setting the corresponding bit of the DDR register to “0”.
  When the pin is configured as an input, the output buffer is turned-off and the pin is put into a high-impedance status.
  When a data is written into the PDR register, the data is retained in the output latch of the PDR, but pin outputs are unaffected.
  Reading the PDR register reads out the pin level (“0” or “1”).

- **Block diagram**



Standby control: Stop, timebase timer mode and SPL=1, or hardware standby mode

(2) N-ch Open-drain Port

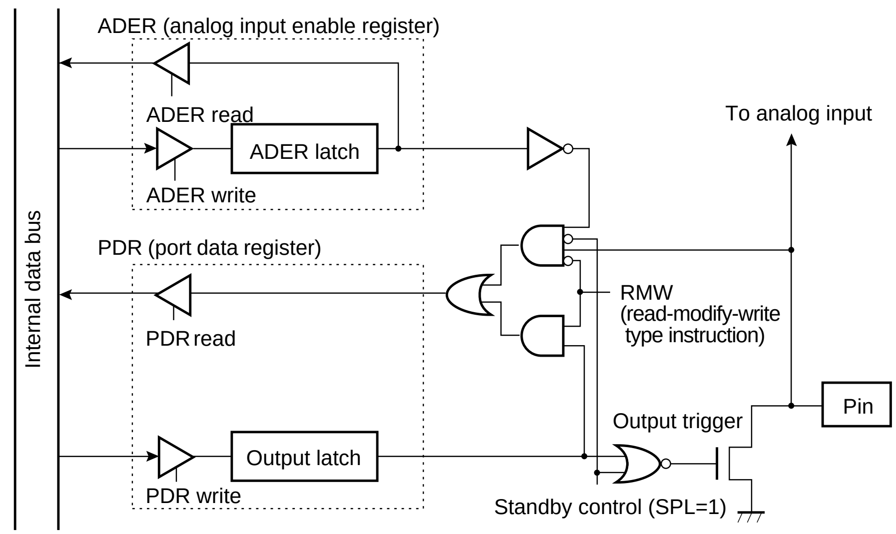
Port 6 is general-purpose I/O port having a combined function as resource input/output. Each pin can be switched between resource and port bitwise.

- Operation as output port
  When a data is written into the PDR register, the data is latched to the output latch of PDR. When the output latch value is set to “0”, the output transistor is turned on and the pin status is put into an “L” level output, while writing “1” turns off the transistor and put the pin in a high-impedance status.
  If the output pin is pulled-up, setting output latch value to “1” puts the pin in the pull-up status.
  Reading the PDR register returns the pin value (same as the output latch value in the PDR).

Note: Execution of a read-modify-write instruction (e.g. bit set instruction) reads out the output latch value rather than the pin value, leaving output latch that is not manipulated unchanged.

- Operation as input port
  Setting corresponding bit of the PDR register to “1” turns off the output transistor and the pin is put into a high-impedance status.
  Reading the PDR register returns the pin value (“0” or “1”).

- **Block diagram**



Standby control: Stop, timebase timer mode and SPL=1, or hardware standby mode

(3) Register Configuration

Port 1 data register (PDR1) — Address 000001H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
P17	P16	P15	P14	P13	P12	P11	P10	(System reservation area)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Port 4 data register (PDR4) — Address 000004H

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(PDR5)	P47	P46	P45	P44	P43	P42	P41	P40
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Port 5 data register (PDR5) — Address 000005H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
P57	P56	P55	P54	P53	P52	P51	P50	(PDR4)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Port 6 data register (PDR6) — Address 000006H

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(PDR7)	P67	P66	P65	P64	P63	P62	P61	P60
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Port 7 data register (PDR7) — Address 000007H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
—	P76	P75	P74	P73	P72	P71	P70	(PDR6)
—	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Port 8 data register (PDR8) — Address 000008H

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(PDR9)	P87	P86	P85	P84	P83	P82	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	—	—

Port 9 data register (PDR9) — Address 000009H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
—	P96	P95	P94	P93	P92	P91	P90	(PDR8)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Port A data register (PDRA) — Address 00000AH

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(Vacancy)	—	—	PA5	PA4	PA3	PA2	PA1	PA0
	—	—	R/W	R/W	R/W	R/W	R/W	R/W

Port 1 direction register (DDR1) — Address 000011H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
P17	P16	P15	P14	P13	P12	P11	P10	(System reservation area)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Port 4 direction register (DDR4) — Address 000014H

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(DDR5)	P47	P46	P45	P44	P43	P42	P41	P40
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Port 5 direction register (DDR5) — Address 000015H

bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0
P57	P56	P55	P54	P53	P52	P51	P50	(DDR4)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Analog input enable register (ADER) — Address 000016H

bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
(DDR7)	P67	P66	P65	P64	P63	P62	P61	P60
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

(Continued)

(Continued)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	
000017H	—	P76	P75	P74	P73	P72	P71	P70	(ADER)	Port 7 direction register (DDR7)
	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	
000018H	(DDR9)	P87	P86	P85	P84	P83	P82	—	—	Port 8 direction register (DDR8)
		R/W	R/W	R/W	R/W	R/W	R/W	—	—	

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	
000019H	—	P96	P95	P94	P93	P92	P91	P90	(DDR8)	Port 9 direction register (DDR9)
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	
00001AH	(Vacancy)	—	—	PA5	PA4	PA3	PA2	PA1	PA0	Port A direction register (DDRA)
		—	—	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readble and writable
— : Unused

2. Timebase Timer

The timebase timer is a 18-bit free-run counter (timebase counter) for counting up in synchronization to the internal count clock (divided-by-2 of oscillation) with an interval timer function for selecting an interval time from four types of 2^{13}/HCLK, 2^{15}/HCLK, 2^{17}/HCLK, and 2^{19}/HCLK.

The timebase timer also has a function for supplying operating clocks for the timer output for the oscillation stabilization time or the watchdog timer etc.

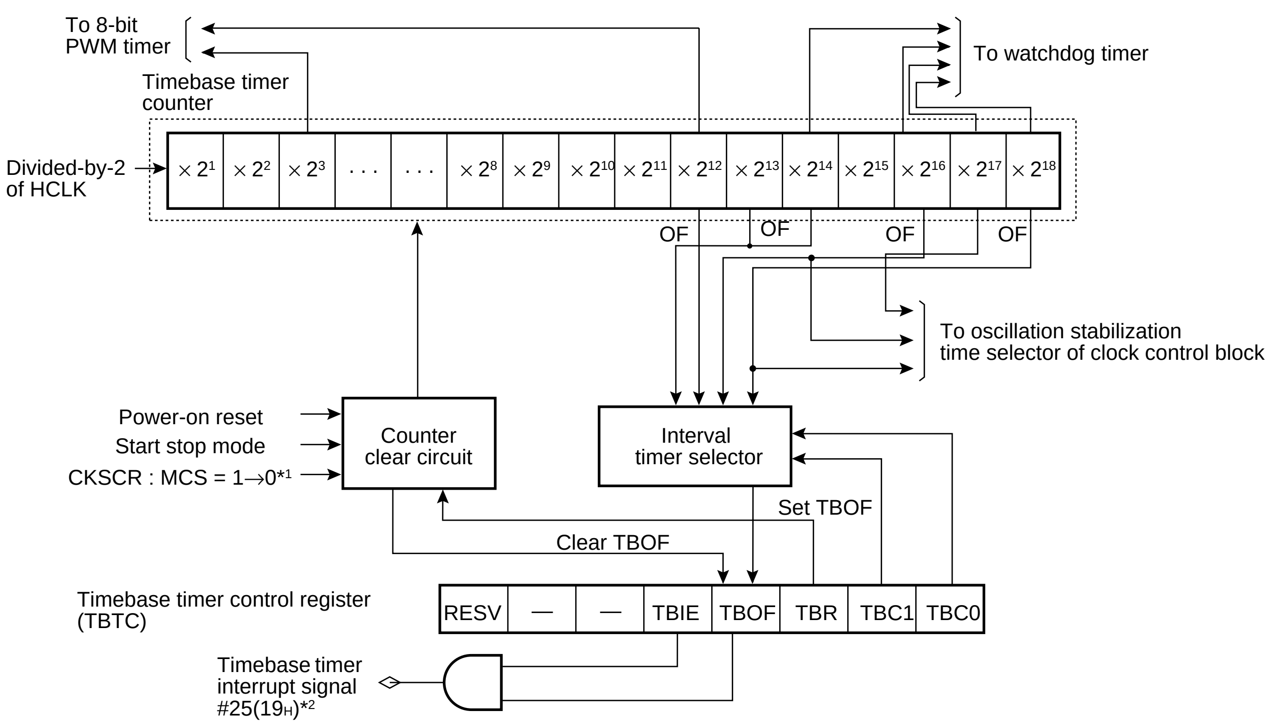
(1) Register Configuration

• Timebase timer control register (TBTC)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············ bit 0	Initial value
0000A9H	RESV	—	—	TBIE	TBOF	TBR	TBC1	TBC0	(WDTC)	0XX00000B
	R/W	—	—	R/W	R/W	W	R/W	R/W		

R/W: Readable and writable
R : Read only
W : Write only
— : Unused
X : Indeterminate
RESV : Reserved bit

(2) Block Diagram



OF : Overflow
HCLK: Oscillation clock
*1 : Switch machine clock from oscillation clock to PLL clock
*2 : Interrupt signal

3. Watchdog Timer

The watchdog timer is a 2-bit counter operating with an output of the timebase timer and resets the CPU when the counter is not cleared for a preset period of time.

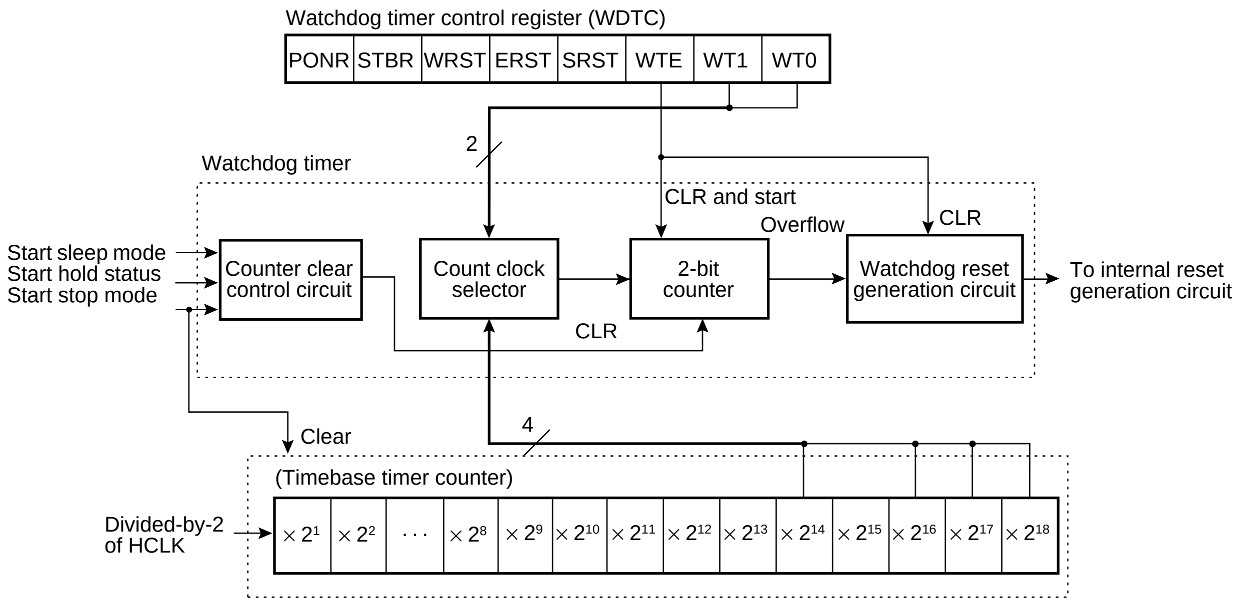
(1) Register Configuration

• Watchdog timer control register (WDTC)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
0000A8H	(TBTC)	PONR	STBR	WRST	ERST	SRST	WTE	WT1	WT0	XXXXXXXXB
		R	R	R	R	R	W	W	W	

R : Read only
W : Write only
X : Indeterminate

(2) Block Diagram



HCLK: Oscillation clock

4. 8-bit PWM Timer

The 8-bit PWM timer is a re-load timer module that can generate a pulse wave with any period/duty ratio. It uses pulse output control according to timer operation for PWM (Pulse Width Modulation) output.

An appropriate external circuit allows the 8-bit PWM timer to operate as a D/A converter.

The 8-bit PWM timer module consists of two 8-bit re-load registers used to specify “H” width and “L” width and of a down counter that is loaded alternately with those values and counts down.

- A pulse waveform with any period and duty ratio is generated.
- An output pulse’s duty ratio of 0.4 to 99.6 percent can be set.
- An appropriate external circuit allows this PWM timer to operate as a D/A converter.
- An interrupt request can be generated by counter underflow.
- The count clock can be selected from two types of timebase timer output.

(1) Register Configuration

• PWM0 to 3 operating mode control register (PWM)

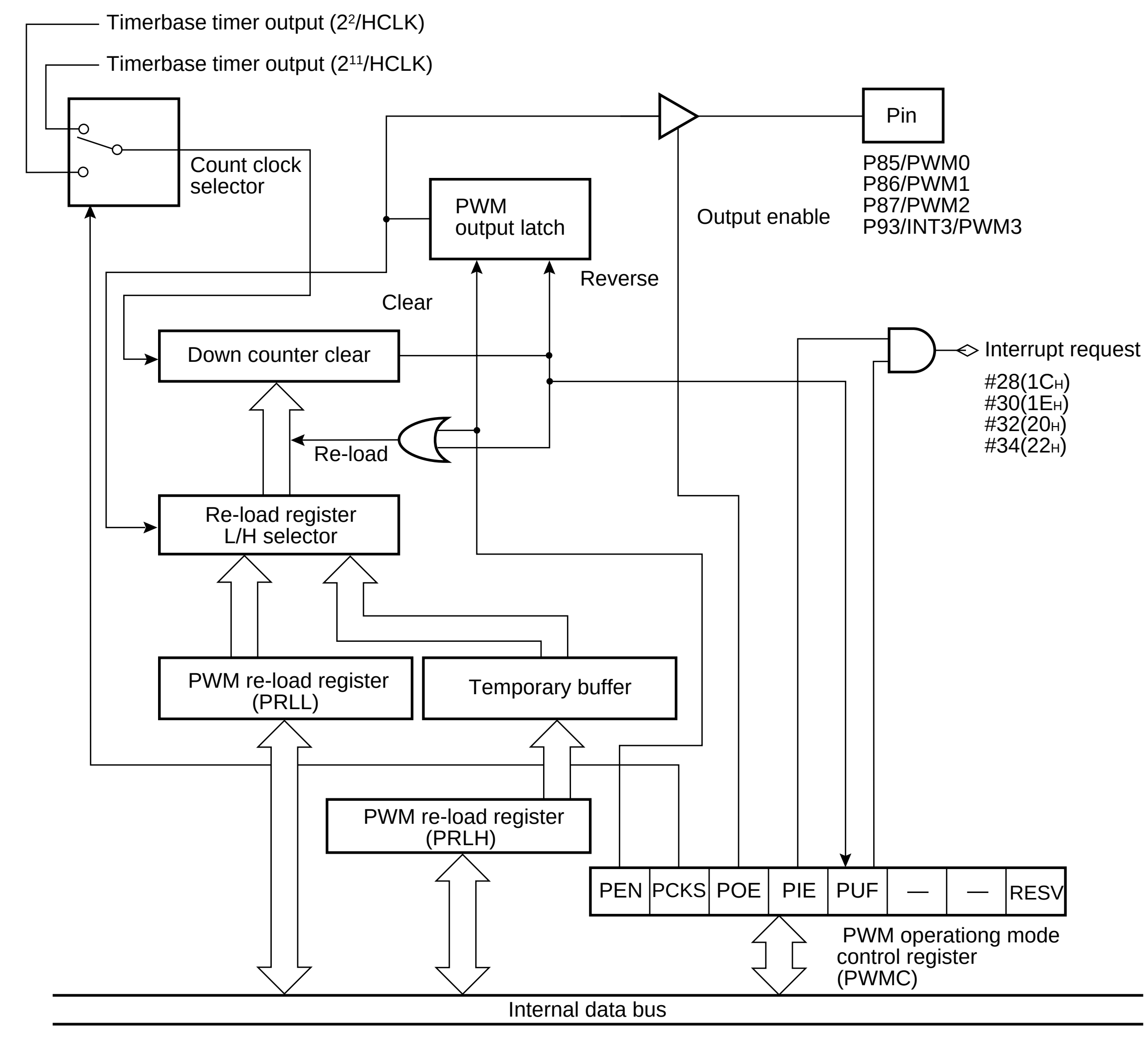
Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
PWMC0 : 000034H	(Vacancy)	PEN	PCKS	POE	PIE	PUF	—	—	RESV	00000XX1B
PWMC1 : 000038H		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	00000XX1B
PWMC2 : 00003CH										00000XX1B
PWMC3 : 00002CH										00000XX1B

• PWM0 to 3 re-load register (PRLL, PRLH)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
PRLH0 : 000037H																	XXXXXXX1B
PRLH1 : 00003BH	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	XXXXXXX1B
PRLH2 : 00003FH																	XXXXXXX1B
PRLH3 : 00002FH																	XXXXXXX1B
PRLL0 : 000036H																	XXXXXXX1B
PRLL1 : 00003AH																	XXXXXXX1B
PRLL2 : 00003EH																	XXXXXXX1B
PRLL3 : 00002EH																	XXXXXXX1B

R/W : Readable and writable
— : Unused
X : Indeterminate
RESV: Reserved bit

(2) Block Diagram



HCLK : Oscillation clock

5. 16-bit Re-load Timer

The 16-bit re-load timer has an internal clock mode for counting down in synchronization to three types of internal clocks and an event count mode for counting down detecting a given edge of the pulse input to the external bus pin, and either of the two functions can be selectively used.

For this timer, an “underflow” is defined as the timing of transition from the counter value of “0000_H” to “$FFFF_H$”. According to this definition, an underflow occurs after [re-load register setting value + 1] counts.

In operating the counter, the re-load mode for repeating counting operation after re-loading a counter value after an underflow or the one-shot mode for stopping the counting operation after an underflow can be selectively used.

Because the timer can generate an interrupt upon an underflow, the timer conforms to the extended intelligent I/O service (EI²OS).

The MB90246A series has 3 channels of 16-bit re-load timers.

(1) Register Configuration

• Timer control status register 0, 1, 2 upper digits (TMCSR0, TMCSR1, TMCSR2: H)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
TMCSR0 : 000041_H TMCSR1 : 000049_H TMCSR2 : 000051_H	—	—	—	—	CSL1	CSL0	MOD2	MOD1	(TMCSR : L)	$----0000_B$
	—	—	—	—	R/W	R/W	R/W	R/W		

• Timer control status register 0, 1, 2 lower digits (TMCSR0, TMCSR1, TMCSR2: L)

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
TMCSR0 : 000040_H TMCSR1 : 000048_H TMCSR2 : 000050_H	(TMCSR : H)	MOD0	OUTE	OUTL	RELD	INTE	UF	CNTE	TRG	00000000_B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

• 16-bit timer register 0, 1 (TMR0, TMR1, TMR2)

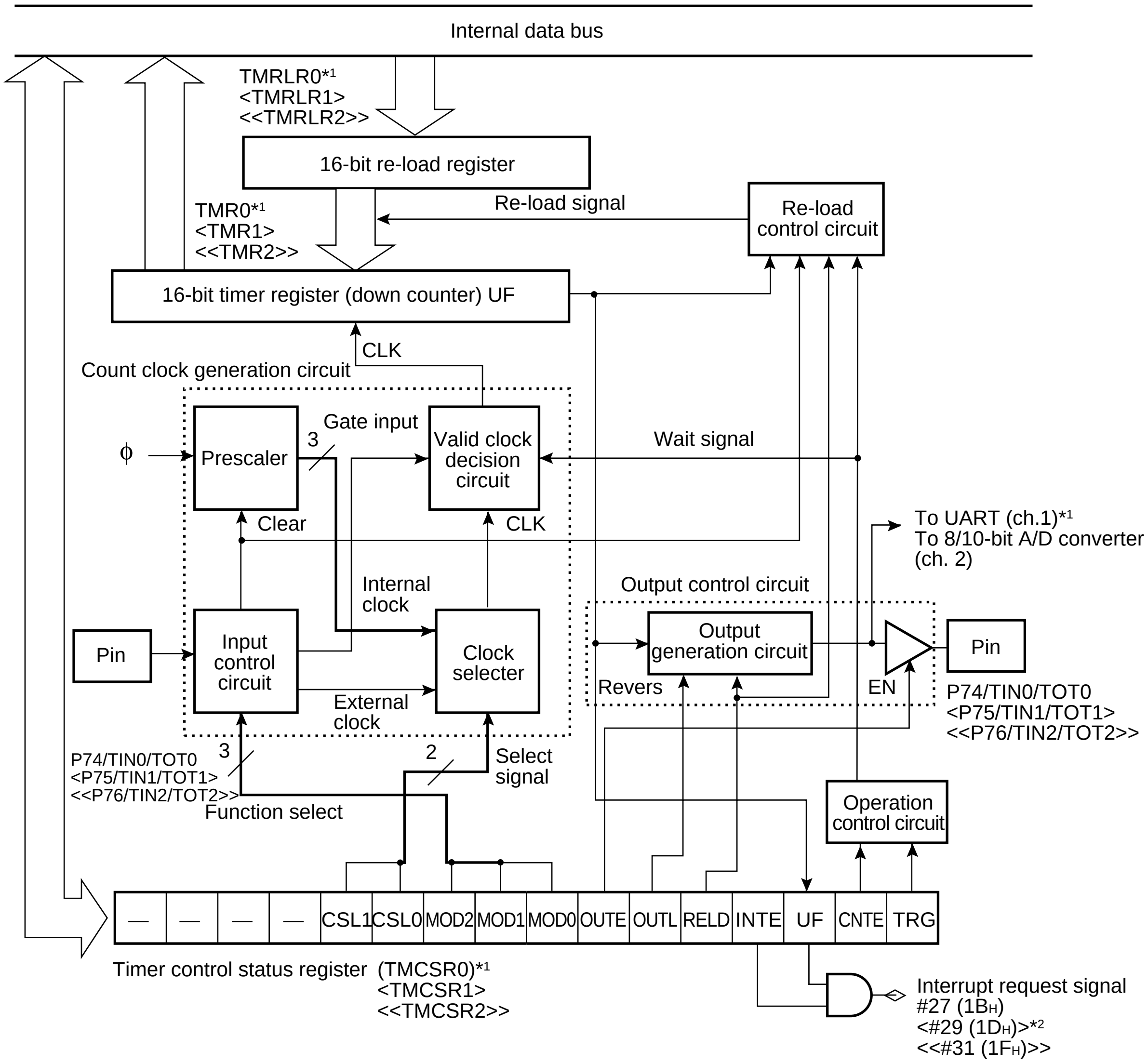
Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
TMR0 : 000042_H TMR1 : $00004A_H$ TMR2 : 000052_H	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	$XXXXXXXX_B$ $XXXXXXXX_B$ $XXXXXXXX_B$
	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

• 16-bit re-load register 0, 1 (TMRL0,TMRL1)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
TMRLR0 : 000044_H TMRLR1 : $00004C_H$ TMRLR2 : 000054_H	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	$XXXXXXXX_B$ $XXXXXXXX_B$ $XXXXXXXX_B$
	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	

R/W : Readable and writable
R : Read only
W : Write only
— : Unused
X : Indeterminate

(2) Block Diagram



*1: The timer has ch.0, ch.1 and ch.2, and listed in the parenthesis <> are for ch.1 and << >> for ch.2.
*2: Interrupt number
ϕ: Machine clock frequency

6. 16-bit I/O Timer

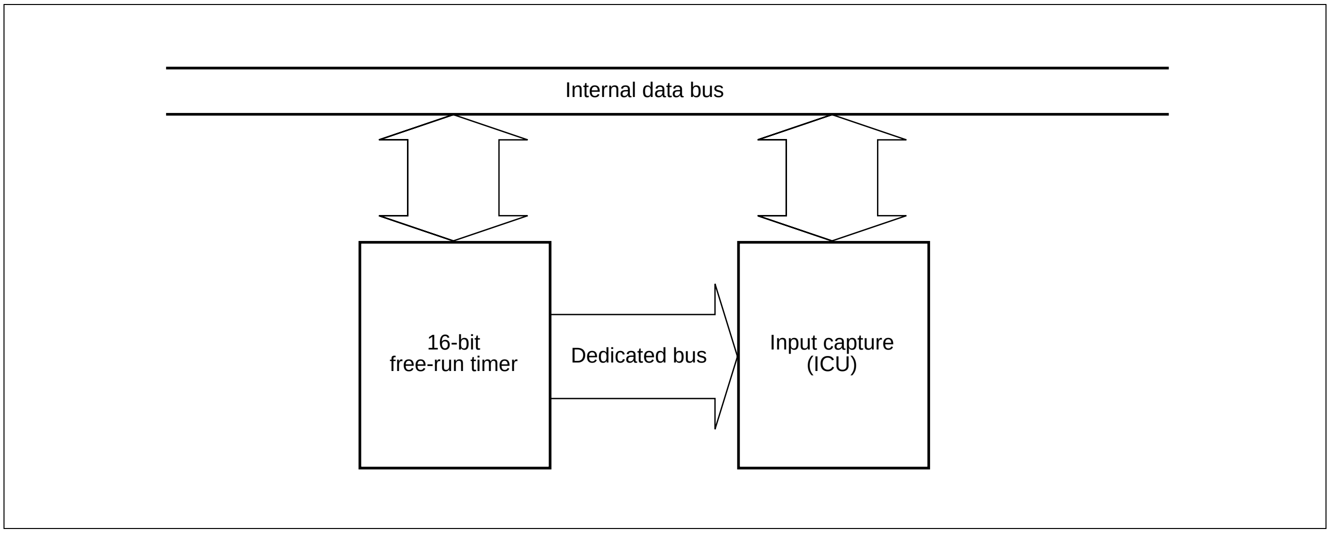
The 16-bit I/O timer module consists of one 16-bit free-run timer, two input capture (ICU) circuits, and four output comparators.

This complex module allows two independent waveforms to be output on the basis of the 16-bit free-run timer. Input pulse width and external clock periods can, therfore, be measured.

The 16-bit I/O timer consists of:

- a 16-bit free-run timer; and
- two input captures (ICU).

- **Block diagram**



(1) 16-bit Free-run Timer

The 16-bit free-run timer consists of a 16-bit up counter, a prescaler, and a control register. The value output from the timer counter is used as basic timer (base timer) for input capture (ICU).

- A counter operation clock can be selected from four internal clocks.
- An interrupt request can be issued to the CPU by counter overflow.
- The extended intelligent I/O service (EI²OS) can be activated.
- The 16-bit free-run timer counter is cleared to “0000H” by a reset or by clearing the timer (TCCS: CLK = 0).

- **Register configuration**

- Timer control status register (TCCS)

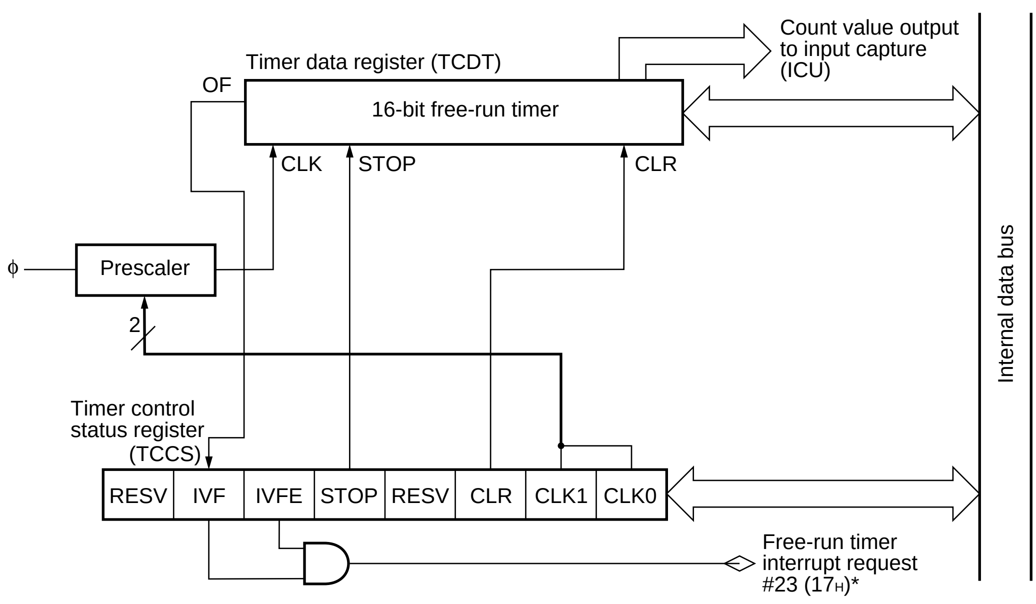
Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
00006EH	(Vacancy)	RESV	IVF	IVFE	STOP	RESV	CLR	CLK1	CLK0	00000000B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

- Timer data register (TCDT)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
00006DH 00006CH	T15	T14	T13	T12	T11	T10	T09	T08	T07	T06	T05	T04	T03	T02	T01	T00	00000000B 00000000B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
RESV : Reserved bit

- **Block diagram**



ϕ : Machine clock frequency
OF : Overflow
* : Interrupt number

(2) Input Capture (ICU)

The input capture (ICU) consists of a capture register corresponding to two 16-bit external input pins, a control register, and an edge detector. Upon input of a trigger edge through an external input pin, the counter value of the 16-bit free-run timer is stored into the input capture register, and an interrupt request can be generated concurrently.

- A capture interrupt can be generated independently for each capture unit.
- The extended intelligent I/O service (EI²OS) can be activated.
- A trigger edge direction can be selected from rising/falling/both edges.
- Since two input capture units can be operated independent of each other, up to two events can be measured independently.
- The input capture function is suited for measurements of intervals (frequencies) and pulse-widths.

- **Register configuration**

- Input capture control status register (ICS)

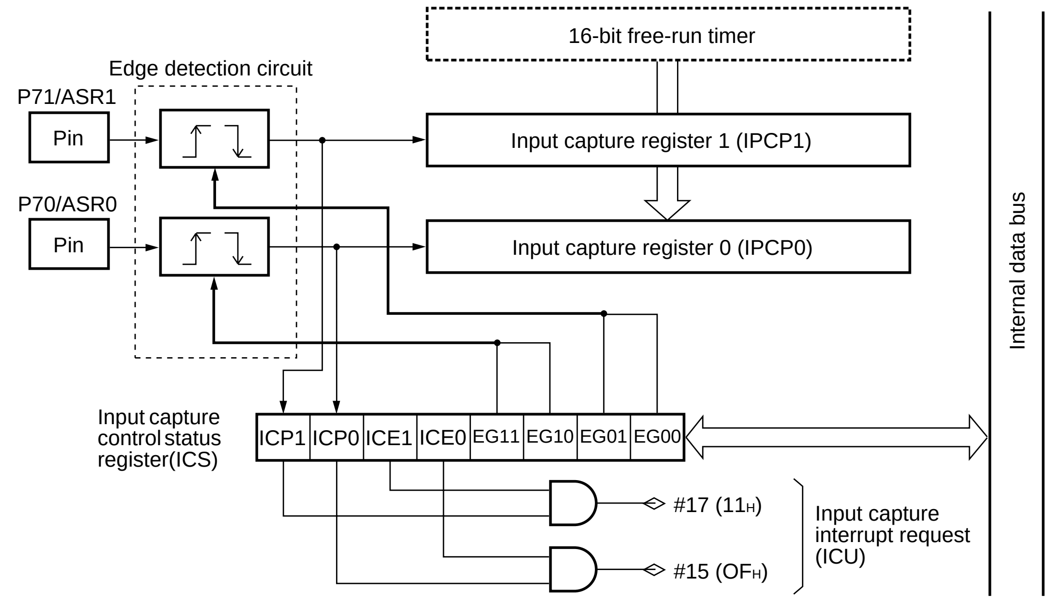
Address	bit 15 · · · · · · · · · · · · bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
ICS0 : 000064H	(Vacancy)	ICP1	ICP0	ICE1	ICE0	EG11	EG10	EG01	EG00	00000000B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

- Input capture register (IPCP0, IPCP1)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
IPCP0 : 000061H IPCP1 : 000063H IPCP0 : 000060H IPCP1 : 000062H	CP15	CP14	CP13	CP12	CP11	CP10	CP09	CP08	CP07	CP06	CP05	CP04	CP03	CP02	CP01	CP00	XXXXXXXXB XXXXXXXXB
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
R : Read only
X : Indeterminate

- **Block diagram**



*: Interrupt number

7. Simple I/O Serial Interface

The 8/16-bit simple I/O serial interface transfers data synchronously with a clock.

- Communications direction: Concurrent processing of transmission (Whether data is to be sent or received must be judged by the user.)
- Transfer mode: Clock synchronization function (Only data are transferred.)
- Transfer rate: DC to $\phi/2$ (ϕ: Machine clock. Frequencies of up to 8 MHz are available when the machine clock is rated at 16 MHz.)
- Shift clock: A machine clock division clock is used as the shift clock. (One of four division ratios can be selected.). A shift clock is output only during data transfer.
- Data transfer format: MSB first can be selected. 8 or 16 bits can be selected as data length. Only data are transferred.
- Interrupt request: An interrupt request is issued upon termination of transfer.
- Inter-CPU connection: Only 1:1 (bidirectional communication)

(1) Register Configuration

- Serial control status register 1, 2 (SCR)

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
SCR0 : 000020H SCR1 : 000024H	(SSR)	STOP	OCKE	SOE	SIE	SIR	WBS	SMD1	SMD0	10000000B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

- Serial status register 1, 2 (SSR)

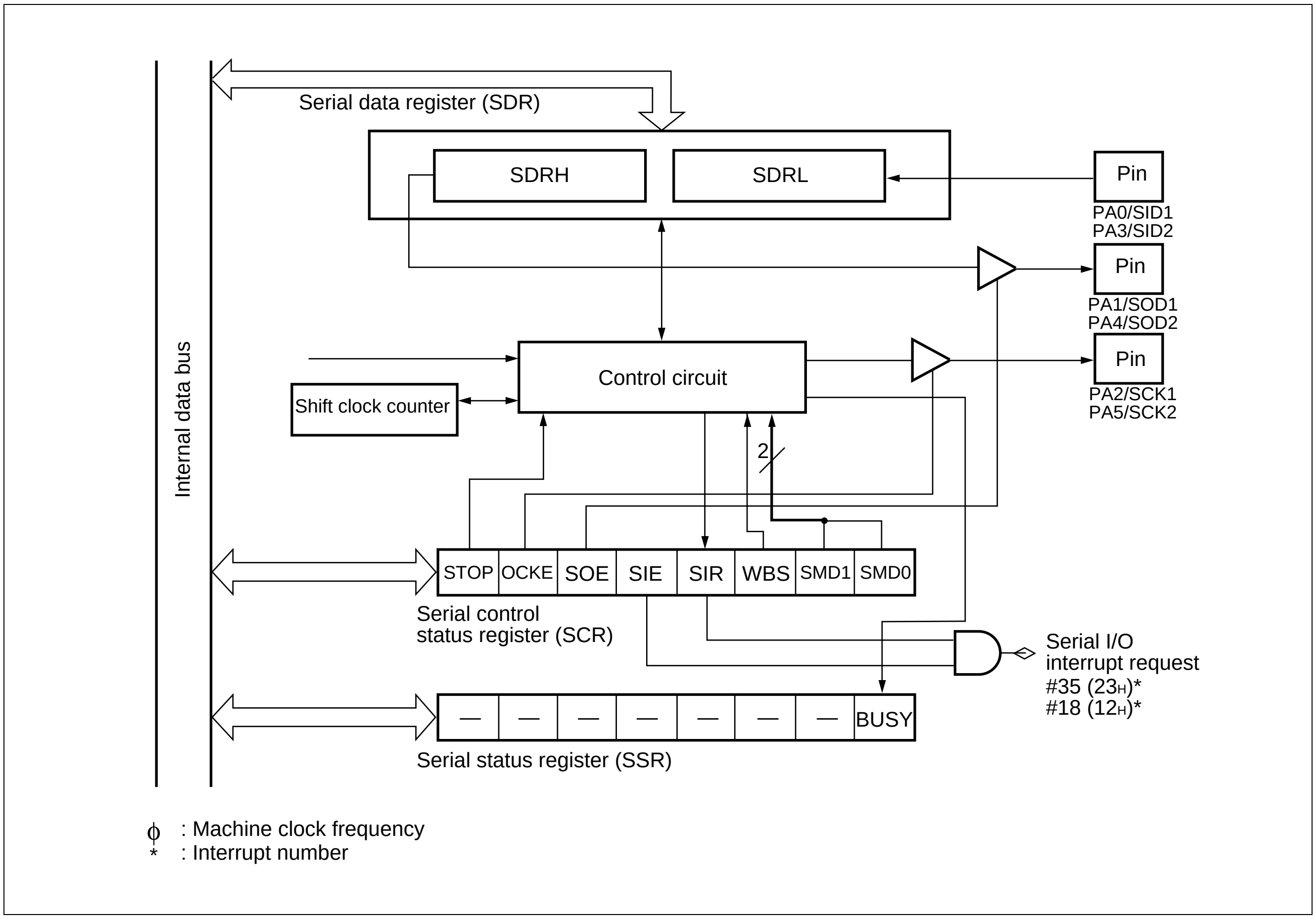
Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
SSR1 : 000021H SSR2 : 000025H	—	—	—	—	—	—	—	BUSY	(SCR)	-------1B
	—	—	—	—	—	—	—	R		

- Serial data register 1, 2 (SDR)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
SDR1H : 000023H SDR2H : 000027H SDR1L : 000022H SDR2L : 000026H	D15	D14	D13	D12	D11	D10	D09	D08	D07	D06	D05	D04	D03	D02	D01	D00	XXXXXXXXB XXXXXXXXB
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
R : Read only
— : Unused
X : Indeterminate

(2) Block Diagram



8. UART

UART0 is a general-purpose serial data communication interface for performing synchronous or asynchronous communication (start-stop synchronization system). In addition to the normal duplex communication function (normal mode), UART0 has a master-slave type communication function (multi-processor mode).

- Data buffer: Full-duplex double buffer
- Transfer mode:Clock synchronized (with start and stop bit)
  Clock asynchronized (start-stop synchronization system)
- Baud rate: With dedicated baud rate generator, selectable from 12 types
  External clock input possible
  Internal clock (A clock supplied from 16-bit re-load timer 2 can be used.)
- Data length: 7 bit to 9 bit selective (with a parity bit)
  6 bit to 8 bit selective (without a parity bit)
- Signal format: NRZ (Non Return to Zero) system
- Reception error detection: Framing error
  Overrun error
  Parity error (not available in multi-processor mode)
- Interrupt request: Receive interrupt (receive complete, receive error detection)
  Receive interrupt (transmit complete)
  Transmit/receive conforms to extended intelligent I/O service (EI^2OS)
- Master/slave type communication function: 1 (master) to n (slave) communication possible (multi-processor mode)

(1) Register Configuration

- Status register (USR)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············· bit 0	Initial value
000029_H	RDRF	OREF	PE	TDRE	RIE	BCH0	RBF	TBF	(UMC)	00010000_B
	R	R	R	R	R/W	R/W	R	R		

- Mode control register (UMC)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000028_H	(USR)	PEN	SBL	MC1	MC0	SMDE	RFC	SCKE	SOE	00000100_B
		R/W	R/W	R/W	R/W	R/W	W	R/W	R/W	

- Rate and data register (URD)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············· bit 0	Initial value
$00002B_H$	BCH	RC3	RC2	RC1	RC0	BCH0	P	D8	(UIDR/UODR)	00000000_B
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

- Input data register (UIDR)

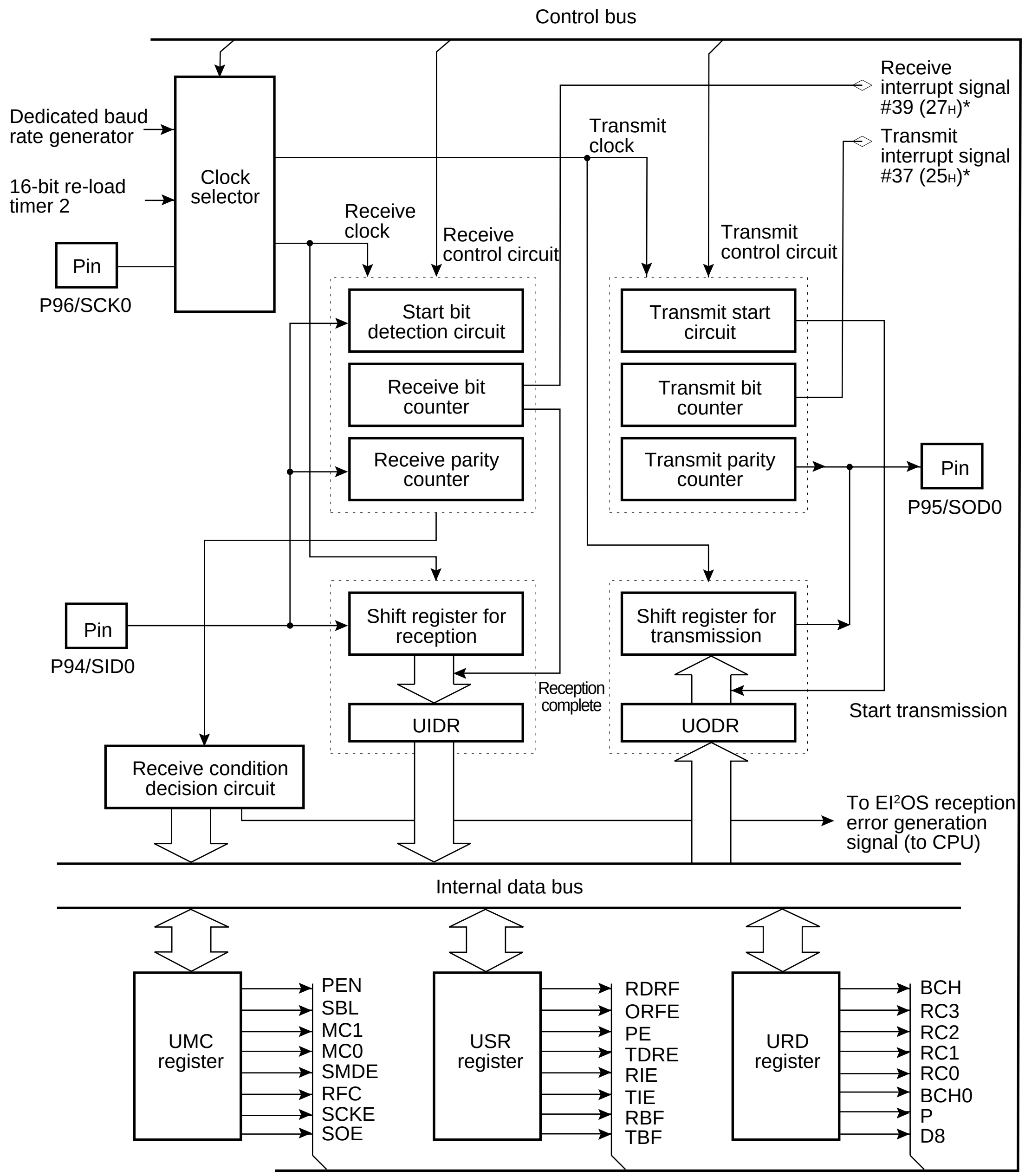
Address	bit 15 ····· bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
$00002A_H$	(URD)	D8	D7	D6	D5	D4	D3	D2	D1	D0	$XXXXXXXX_B$
		R	R	R	R	R	R	R	R	R	

- Output data register (UODR)

Address	bit 15 ····· bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
$00002A_H$	(URD)	D8	D7	D6	D5	D4	D3	D2	D1	D0	$XXXXXXXX_B$
		W	W	W	W	W	W	W	W	W	

R/W : Readable and writable
R : Read only
W : Write only
X : Indeterminate

(2) Block Diagram



* : Interrupt number

9. DTP/External Interrupt Circuit

The DTP (Data Transfer Peripheral)/external interrupt circuit is located between peripheral equipment connected externally and the F²MC-16F CPU and transmit interrupt requests or data transfer requests generated by peripheral equipment to the CPU, generates external interrupt request and starts the extended intelligent I/O service (EI²OS).

(1) Register Configuration

• DTP/interrupt factor register (EIRR)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············ bit 0	Initial value
000031_H	RESV	RESV	RESV	RESV	ER3	ER2	ER1	ER0	(ENIR)	----0000$_B$
	—	—	—	—	R/W	R/W	R/W	R/W		

• DTP/interrupt enable register (ENIR)

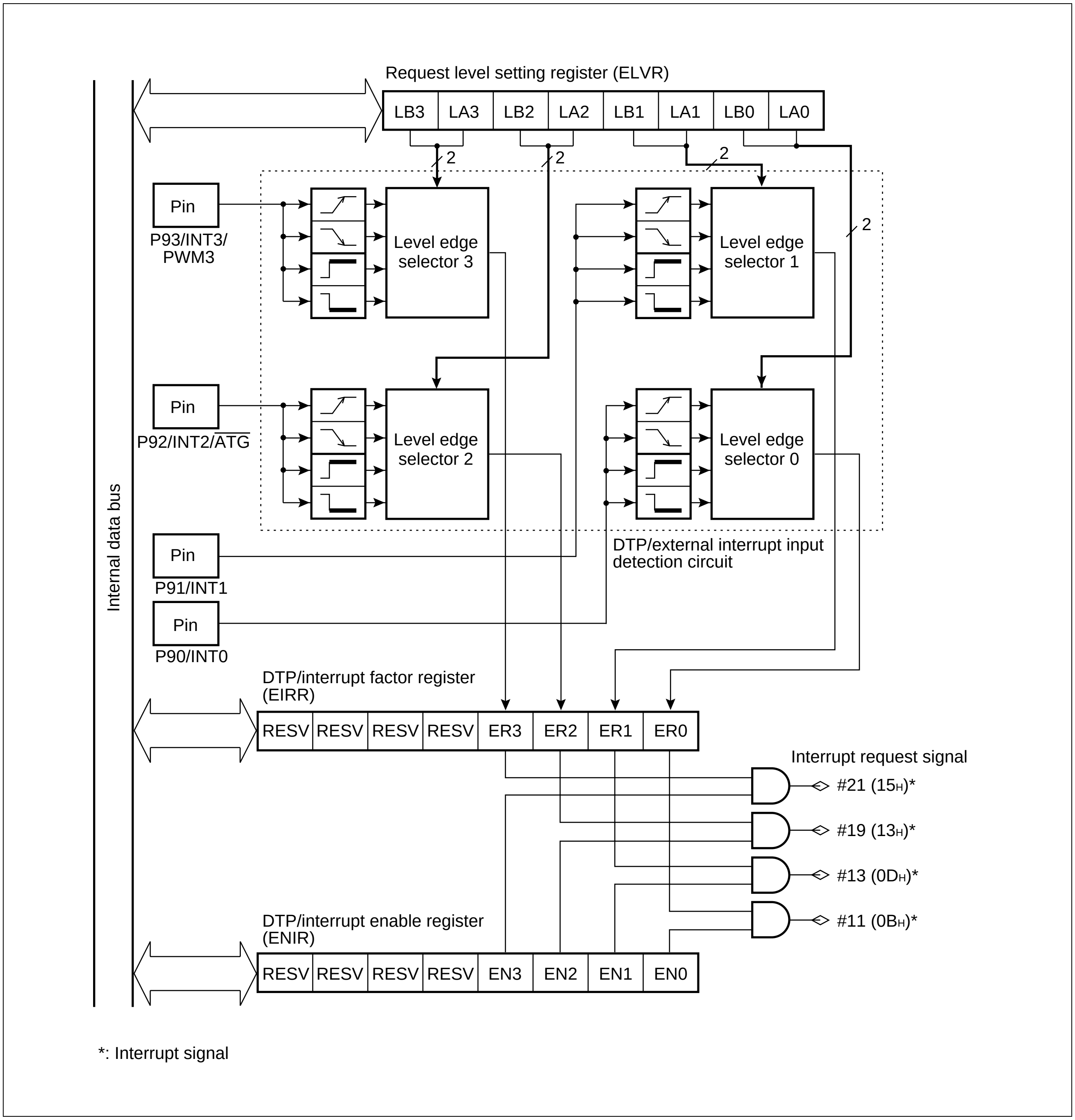
Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000030_H	(EIRR)	RESV	RESV	RESV	RESV	EN3	EN2	EN1	EN0	----0000$_B$
		—	—	—	—	R/W	R/W	R/W	R/W	

• Request level setting register (ELVR)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000032_H	(Vacancy)	LB3	LA3	LB2	LA2	LB1	LA1	LB0	LA0	00000000$_B$
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
— : Unused
RESV : Reserved bit

(2) Block Diagram



10. Delayed Interrupt Generation Module

The delayed interrupt generation module generates interrupts for switching tasks for development on a real-time operating system (REALOS series). The module can be used to generate softwarewise generates hardware interrupt requests to the CPU and cancel the interrupts.

This module does not conform to the extended intelligent I/O service (EI^2OS).

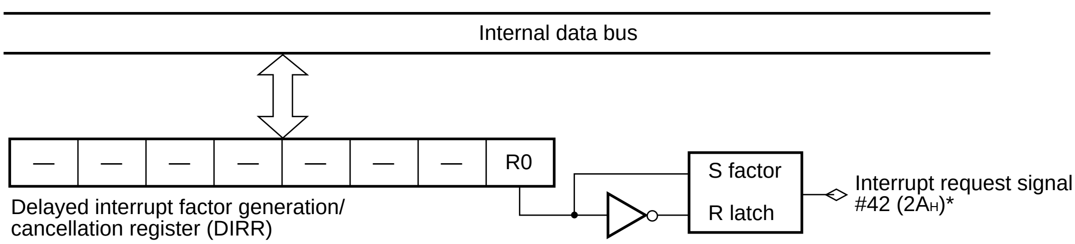
(1) Register Configuration

• Delayed interrupt factor generation/cancellation register (DIRR)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············ bit 0	Initial value
$00009F_H$	—	—	—	—	—	—	—	R0	(System reservation area)	- - - - - - - 0 $_B$
	—	—	—	—	—	—	—	R/W		

R/W : Readable and writable
— : Unused

(2) Block Diagram



*: Interrupt signal

11. 8/10-bit A/D Converter

The 8/10-bit A/D converter has a function of converting analog voltage input to the analog input pins (input voltage) to digital values (A/D conversion) and has the following features.

- Minimum conversion time: 6.13 µs (at machine clock of 16 MHz, including sampling time)
- Minimum sampling time: 3.75 µs (at machine clock of 16 MHz)
- Conversion time: The sampling time can be set arbitrarily.
  Serial to parallel converter with a sample hold circuit
- Conversion method
- Resolution: 10-bit or 8-bit selective
- Analog input pins: Selectable from eight channels by software
  Single conversion mode: Single conversion for the specified channel
  Scan conversion mode: Scan conversions for maximum of four channel
- Interrupt requests can be generated and the extended intelligent I/O service (EI^2OS) can be started after the end of A/D conversion.
- Starting factors for conversion: Selected from software activation, 16-bit re-load timer 1 output (rising edge), and external trigger (falling edge).
- A data buffer that covers four channels is supported. The results of conversion are stored into the data buffer.

(1) Register Configuration

• A/D control status register upper digits (ADCSH)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ············ bit 0	Initial value
000071H	—	ACS2	ACS1	ACS0	—	—	CREG	SCAN	(ADCSL)	-000--00B
	—	R/W	R/W	R/W	—	—	R/W	R/W		

• A/D control status register lower digits (ADCSL)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000070H	(ADCSH)	BUSY	INT	INTE	—	STS1	STS0	STAR	RESV	000-0000B
		R/W	R/W	R/W	—	R/W	R/W	R/W	R/W	

• A/D data register 0 to 3 (ADTH, ADTL)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
ADTH0 : 000075H ADTH1 : 000077H ADTH2 : 000079H ADTH3 : 00007BH ADTL0 : 000074H ADTL1 : 000076H ADTL2 : 000078H ADTL3 : 00007AH	—	—	—	—	—	—	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	------**B XXXXXXXXB
	R	R	R	R	R	R	*	*	R	R	R	R	R	R	R	R	

• Conversion time setting register (ADCT)

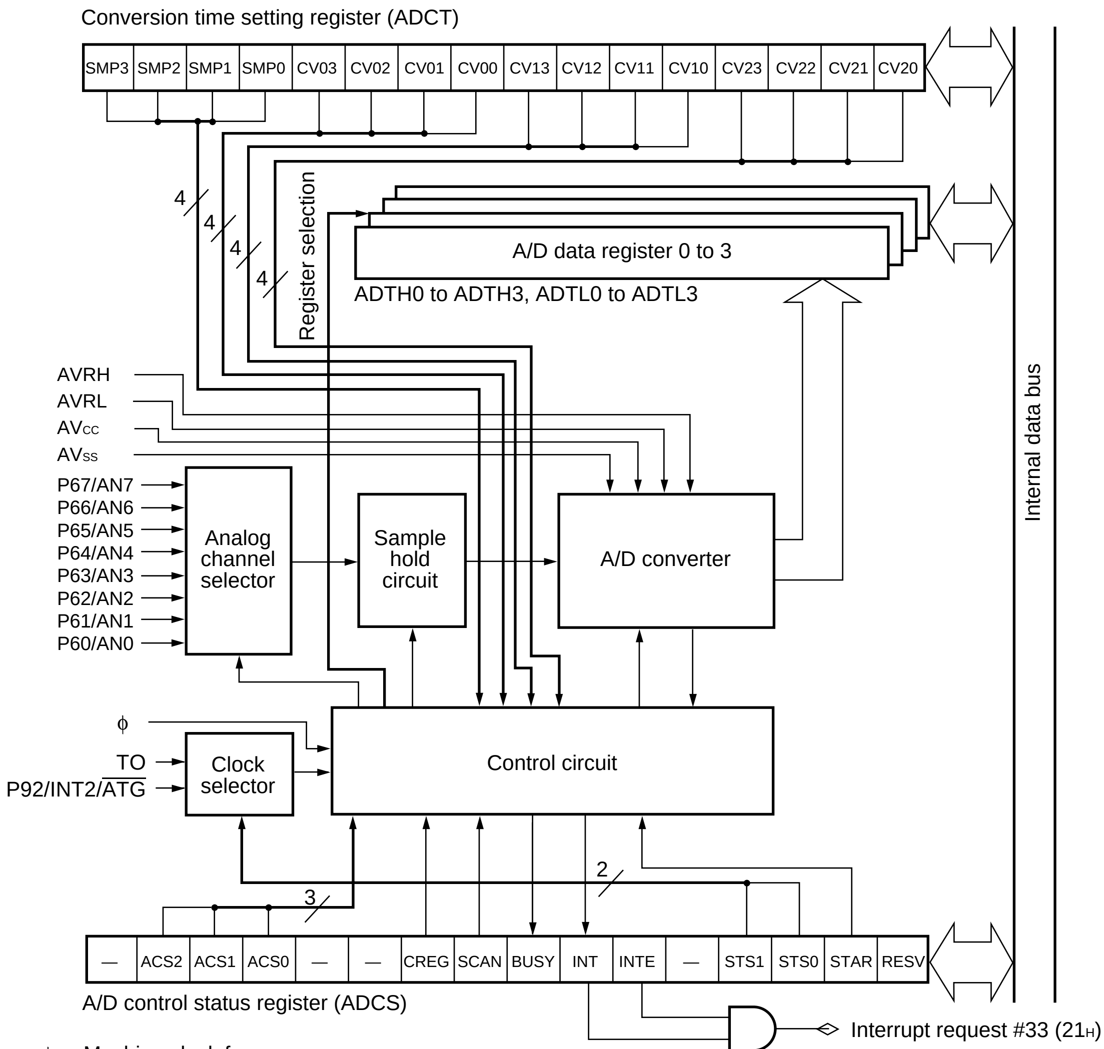
Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000073H 000072H	SMP3	SMP2	SMP1	SMP0	CV03	CV02	CV01	CV00	CV13	CV12	CV11	CV10	CV23	CV22	CV21	CV20	XXXXXXXXB XXXXXXXXB
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

• Analog input enable register (ADER)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000016H	(DDR7)	ADE7	ADE6	ADE5	ADE4	ADE3	ADE2	ADE1	ADE0	11111111B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W: Readable and writable
R : Read only
— : Unused
X : Indeterminate
* : The CREG bit value of ADCSH makes different storage styles.
RESV : Reserved bit

(2) Block Diagram



ϕ : Machine clock frequency
TO : 16-bit re-load timer channel 1 output

12. 8-bit D/A Converter

The 8-bit D/A converter, which is based on the R-2R system, supports 8-bit resolution mode. It contains two channels each of which can be controlled in terms of output by the D/A control register.

(1) Register Configuration

• D/A control register 0 (DACR0)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
00005BH	—	—	—	—	—	—	—	DAE0	(DADR0)	- - - - - - - 0B
	—	—	—	—	—	—	—	R/W		

• D/A control register 1 (DACR1)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
00005DH	—	—	—	—	—	—	—	DAE1	(DADR1)	- - - - - - - 0B
	—	—	—	—	—	—	—	R/W		

• D/A control register 2 (DACR2)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
00005FH	—	—	—	—	—	—	—	DAE2	(DADR2)	- - - - - - - 0B
	—	—	—	—	—	—	—	R/W		

• D/A data register 0 (DADR0)

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
00005AH	(DACR0)	DA07	DA06	DA05	DA04	DA03	DA02	DA01	DA00	XXXXXXXXB
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

• D/A data register 1 (DADR1)

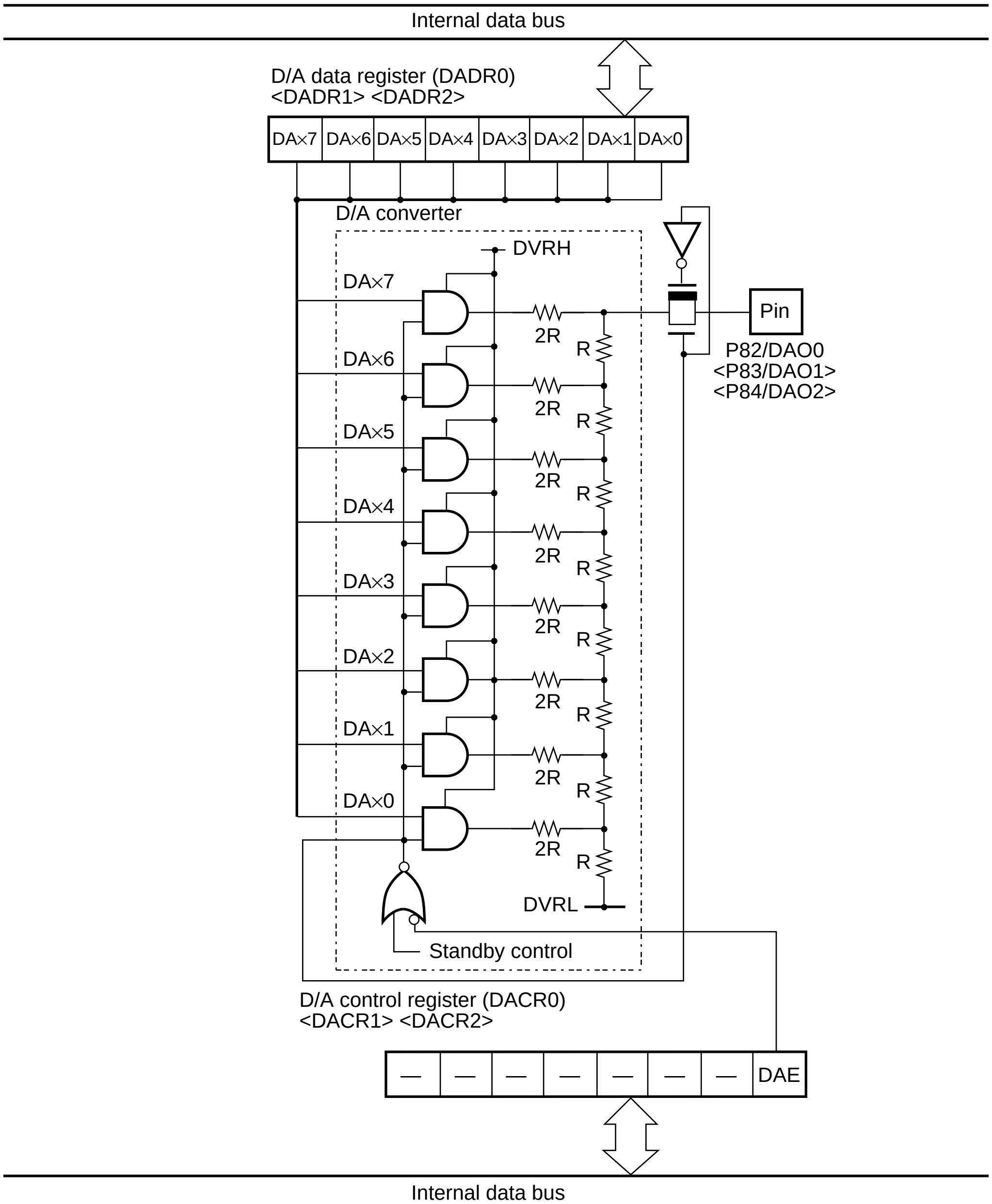
Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
00005CH	(DACR1)	DA17	DA16	DA15	DA14	DA13	DA12	DA11	DA10	XXXXXXXXB
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

• D/A data register 2 (DADR2)

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
00005EH	(DACR2)	DA27	DA26	DA25	DA24	DA23	DA22	DA21	DA20	XXXXXXXXB
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
— : Unused
X : Indeterminate

(2) Block Diagram



Note: The 8-bit D/A converter supports channels 0 to 2. A value enclosed by < and > is for channels 1 and 2.

13. DSP Interface for the IIR Filter

The DSP interface for the IIR filter is a unit which covers product addition (ΣBi × Yj + ΣAm × Xn) by hardware. This interface allows IIR filter calculation to be performed readily and in a high speed.

The DSP interface for the IIR filter has the following features.

- Coefficients A and B, and variables X and Y have 16-bit length, and four banks are supported.
- (1 to 4) + (1 to 4) product terms can be selected.
- Data can be rounded and clipped in units of 10 or 12 bits.
- With two or more concatenated banks used, the results of an operation can be transferred to the subsequent bank register.
- Operation time: ((M + N + 1) × B + 1)/φ µs(M, N = number of product terms, B = number of banks, φ: machine clock)

(1) Register Configuration

- Product addition control status register upper digits (MCSR:H)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
000081H	—	WEY	WENY	WENX	N1	N0	M1	M0	(MCSR:L)	-XXXXXXXB
	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

- Product addition control status register lower digits (MCSR:L)

Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000080H	(MCSR:H)	RND	CLP	DIV	BF	BNK1	BNK0	TRG	MAE	XXX0XXX0B
		R/W	R/W	R/W	R	R/W	R/W	W	R/W	

- Product addition control register upper digits (MCCR:H)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7 ··· bit 0	Initial value
000083H	—	—	—	—	—	—	RESV	RESV	(MCCR:L)	------00B
	—	—	—	—	—	—	R/W	R/W		

- Product addition control register lower digits (MCCR:L)

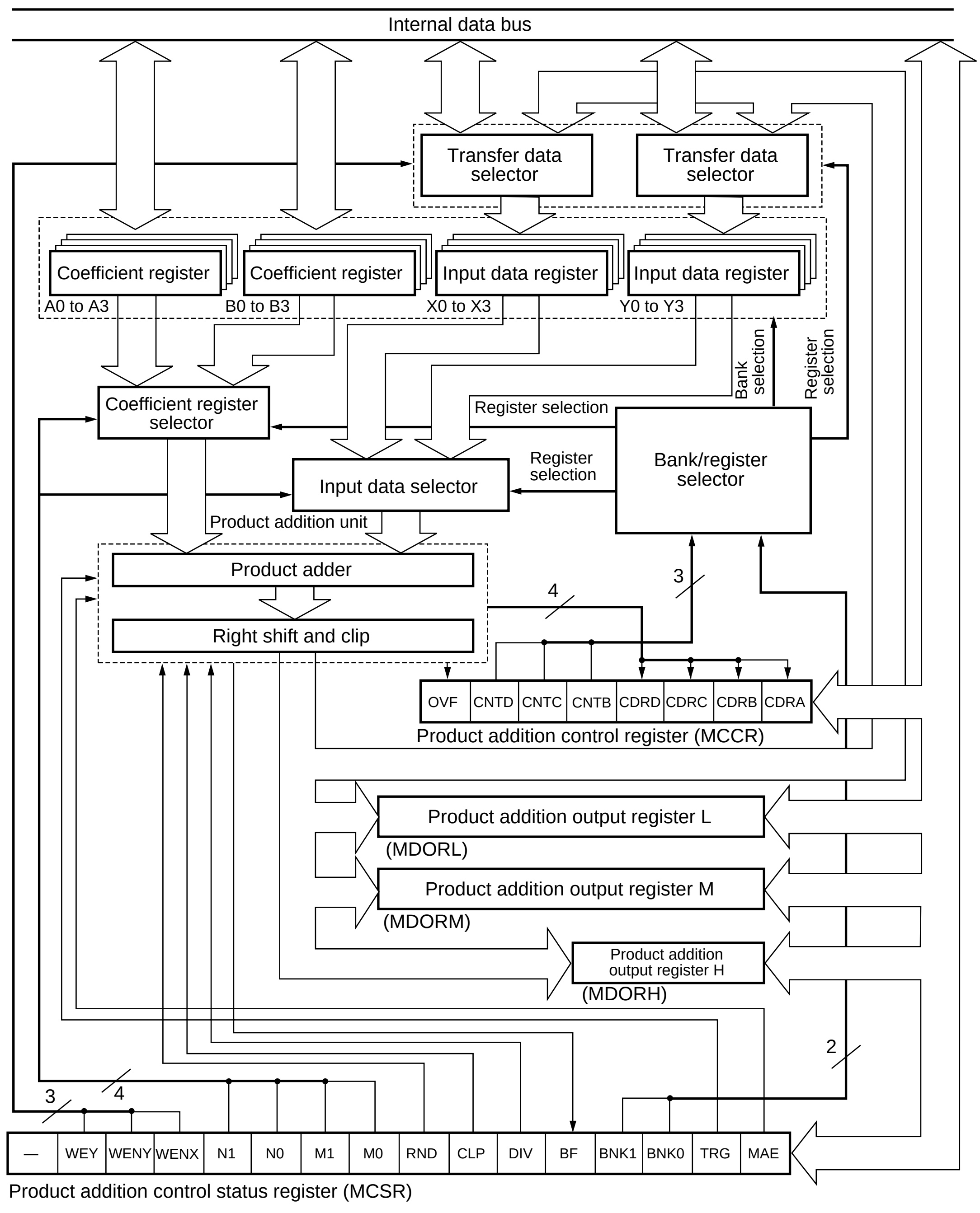
Address	bit 15 ··· bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
000082H	(MCCR:H)	OVF	CNTD	CNTC	CNTB	CDRD	CDRC	CDRB	CDRA	00000000B
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

- Product addition output register (MDORL, M, H)

Address	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
MDORH : 000088H									S	S	S	S	S	D34	D33	D32	XXXXXXXXB
									R	R	R	R	R	R	R	R	
MDORM : 000086H	D31	D30	D29	D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16	XXXXXXXX XXXXXXXXB
	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
MDORL : 000084H	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	XXXXXXXX XXXXXXXXB
	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

R/W: Readable and writable
R : Read only
W : Write only
— : Unused
X : Indeterminate
RESV : Reserved bit

(2) Block Diagram



14. Low-power Consumption (Stand-by) Mode

The F²MC-16F has the following CPU operating mode configured by selection of an clock operation control.

- **Stand-by mode**
  The hardware stand-by mode is a mode for reducing power consumption by stopping clock supply to the CPU by the low-power consumption control circuit, and stopping oscillation clock (stop mode, hardware standby mode).
  Gear function contributes to the low-power dissipation by providing options of divide-by-2, 4, or 16 external clock frequencies, whichiare usually derived from non-divided frequencies.

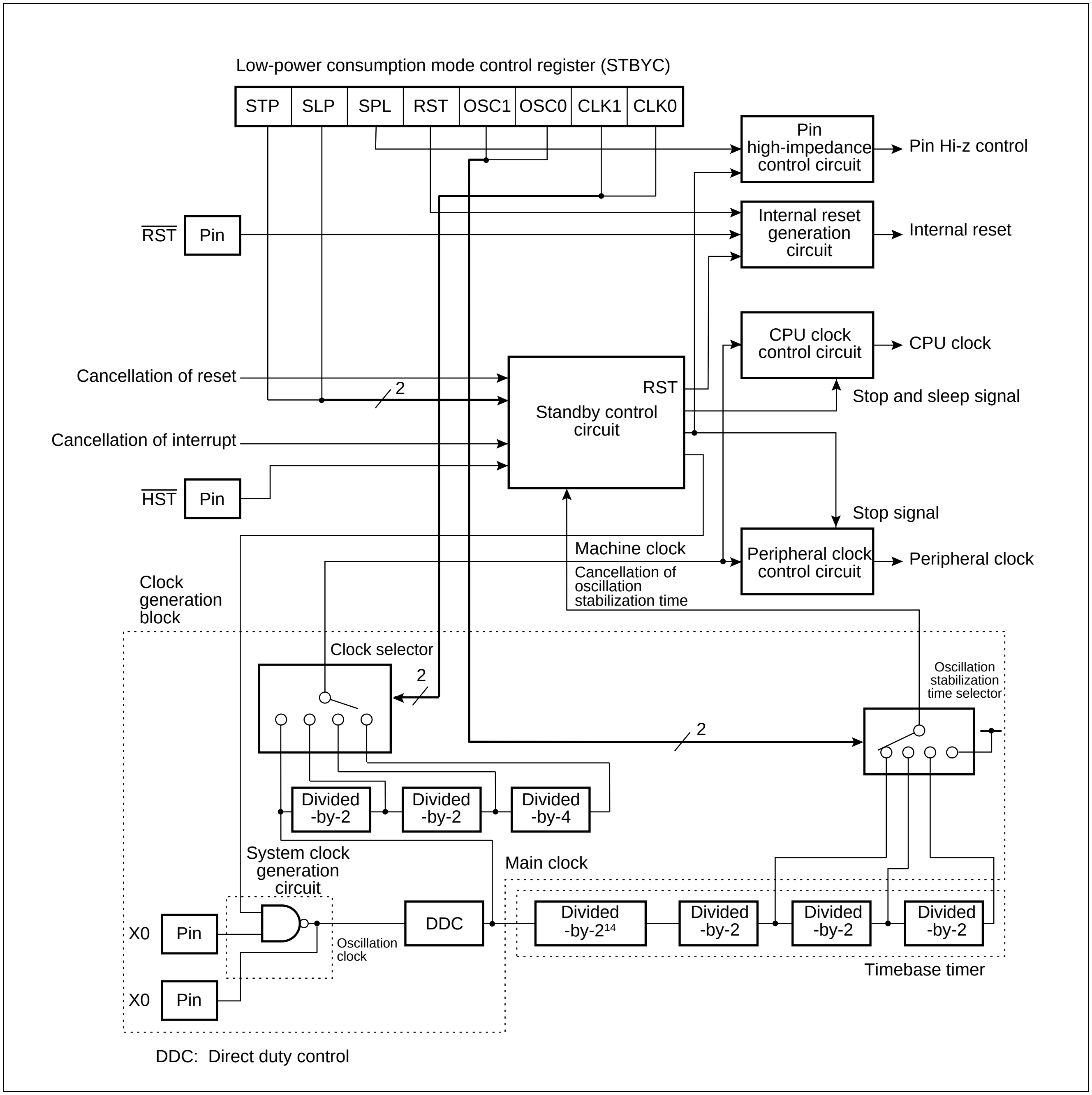
(1) Register Configuration

- Standby control register (STBYC)

Address	bit 15 ············ bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Initial value
$0000A0_H$	(Vacancy)	STP	SLP	SPL	RST	OSC1	OSC0	CLK1	CLK0	$0001XXXX_B$
		W	W	R/W	R/W	R/W	R/W	R/W	R/W	

R/W : Readable and writable
W : Write only
X : Indeterminate

(2) Block Diagram



■ ELECTRICAL CHARACTERISTICS

1. Absolute Maximum Ratings

($AV_{SS} = V_{SS} = 0.0$ V)

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	$V_{SS} – 0.3$	$V_{SS} + 7.0$	V	
	AV_{CC}	$V_{SS} – 0.3$	$V_{SS} + 7.0$	V	*1
	AVRH, AVRL	$V_{SS} – 0.3$	$V_{SS} + 7.0$	V	*1
	DVRH, DVRL	$V_{SS} – 0.3$	$V_{SS} + 7.0$	V	*1
Input voltage	V_I	$V_{SS} – 0.3$	$V_{CC} + 0.3$	V	*2
Output voltage	V_O	$V_{SS} – 0.3$	$V_{CC} + 0.3$	V	*2
“L” level maximum output current	I_{OL}	—	10	mA	*3
“L” level average output current	I_{OLAV}	—	4	mA	*4
“L” level total average output current	ΣI_{OLAV}	—	50	mA	*5
“H” level maximum output current	I_{OH}	—	–10	mA	*3
“H” level average output current	I_{OHAV}	—	–4	mA	*4
“H” level total average output current	ΣI_{OHAV}	—	–48	mA	*5
Power consumption	P_D	—	600	mW	
Operating temperature	T_A	–30	+70	°C	
Storage temperature	Tstg	–55	+150	°C	

*1: AV_{CC}, AVRH, AVRL, DVRH and DVRL shall never exceed V_{CC}.
DVRL shall never exceed DVRH. AVRL shall never exceed AVRH.
*2: V_I and V_O shall never exceed V_{CC} + 0.3 V.
*3: The maximum output current is a peak value for a corresponding pin.
*4: Average output current is an average current value observed for a 100 ms period for a corresponding pin.
*5: Total average current is an average current value observed for a 100 ms period for all corresponding pins.

WARNING: Semiconductor devices can be permanently damaged by application of stress (voltage, current, temperature, etc.) in excess of absolute maximum ratings. Do not exceed these ratings.

2. Recommended Operating Conditions

($AV_{SS} = V_{SS} = 0.0$ V)

Parameter	Symbol	Value		Unit	Remarks
		Min.	Max.		
Power supply voltage	V_{CC}	4.5	5.5	V	Normal operation
	V_{CC}	2.0	5.5	V	Retains RAM data at the time of operation stop
Operating temperature	T_A	–30	+70	°C	External bus mode

WARNING: The recommended operating conditions are required in order to ensure the normal operation of the semiconductor device. All of the device's electrical characteristics are warranted when the device is operated within these ranges.

Always use semiconductor devices within their recommended operating condition ranges. Operation outside these ranges may adversely affect reliability and could result in device failure.

No warranty is made with respect to uses, operating conditions, or combinations not represented on the data sheet. Users considering application outside the listed conditions are advised to contact their FUJITSU representatives beforehand.

3. DC Characteristics

($AV_{CC} = V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –30°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
"H" level input voltage	V_{IH}	CMOS input pin	—	0.7 V_{CC}	—	V_{CC} + 0.3	V	
	V_{IH2}	TTL input pin	V_{CC} = 5.0 V ±10%	2.2	—	V_{CC} + 0.3	V	
	V_{IH1S}	Hysteresis input pin	—	0.8 V_{CC}	—	V_{CC} + 0.3	V	
	V_{IHM}	MD0 to MD2		V_{CC} – 0.3	—	V_{CC} + 0.3	V	
"L" level input voltage	V_{IL1}	CMOS input pin		V_{CC} – 0.3	—	0.3 V_{CC}	V	
	V_{IL2}	TTL input pin	V_{CC} = 5.0 V ±10%	V_{CC} – 0.3	—	0.8	V	
	V_{IL1S}	Hysteresis input pin	—	V_{CC} – 0.3	—	0.2 V_{CC}	V	
	V_{ILM}	MD0 to MD2		V_{CC} – 0.3	—	V_{CC} + 0.3	V	
"H" level output voltage	V_{OH}	All ports other than P60 to P67	V_{CC} = 4.5 V I_{OH} = –4.0 mA	V_{CC} – 0.5	—	—	V	
"L" level output voltage	V_{OL}	All output pins	V_{CC} = 4.5 V I_{OL} = 4.0 mA	—	—	0.4	V	
Open-drain output leakage current	I_{LEAK}	P60 to P67	—	—	0.1	10	µA	
"H" level input current	I_{IH1}	CMOS input pins other than $\overline{RST}$	V_{CC} = 5.5 V V_{IH} = 0.7 V_{CC}	—	—	–10	µA	
	I_{IH2}	TTL input pin	V_{CC} = 5.5 V V_{IH} = 2.2 V_{CC}	—	—	–10	µA	
	I_{IH3}	Hysteresis input pin	V_{CC} = 5.5 V V_{IH} = 0.8 V_{CC}	—	—	–10	µA	
"L" level input current	I_{IL1}	CMOS input pins other than $\overline{RST}$	V_{CC} = 5.5 V V_{IL} = 0.3 V_{CC}	—	—	10	µA	
	I_{IL2}	TTL input pin	V_{CC} = 5.5 V V_{IL} = 0.8 V	—	—	10	µA	
	I_{IL3}	Hysteresis input pin	V_{CC} = 5.5 V V_{IL} = 0.2 V_{CC}	—	—	10	µA	
Pull-up resistance	R	$\overline{RST}$	—	22	—	110	kΩ	

(Continued)

(Continued)

($AV_{CC} = V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –30°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Power supply current	I_{CC}	V_{CC}	Internal operation at 16 MHz V_{CC} = 5.0 V ±10% Normal operation	—	80	100	mA	
	I_{CCS}	—	Internal operation at 16 MHz V_{CC} = 5.0 V ±10% In sleep mode	—	30	50	mA	
	I_{CCH}	—	T_A = +25°C V_{CC} = 4.5 V to 5.5 V In stop mode and hardware standby mode	—	0.1	10	µA	
Input capacitance	C_{IN}	Other than AV_{CC}, AV_{SS}, V_{CC}, V_{SS}	—	—	10	—	pF	

4. AC Characteristics

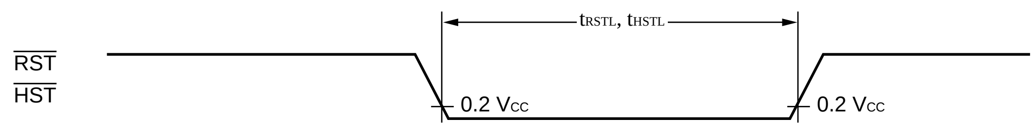
(1) Reset, Hardware Standby Input Timing

($AV_{CC} = V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –30°C to +70°C)

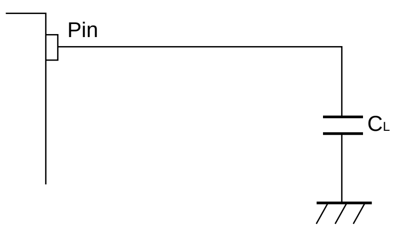
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Reset input time	t_{RSTL}	$\overline{RST}$	—	5 t_{CYC}*	—	ns	
Hardware standby input time	t_{HSTL}	$\overline{HST}$		5 t_{CYC}*	—	ns	

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."

Note: Upon hardware standby input, divide-by-32 is selected as the machine cycle.



- **Measurement conditions for AC ratings**



C_L is a load capacitance connected to a pin under test.

Capacitors of C_L = 30 pF should be connected to CLK pin, while C_L of 80 pF is connected to address bus (A23 to A00) and data bus (D15 to D00), $\overline{RD}$, $\overline{WRH}$ and $\overline{WRL}$ pins.

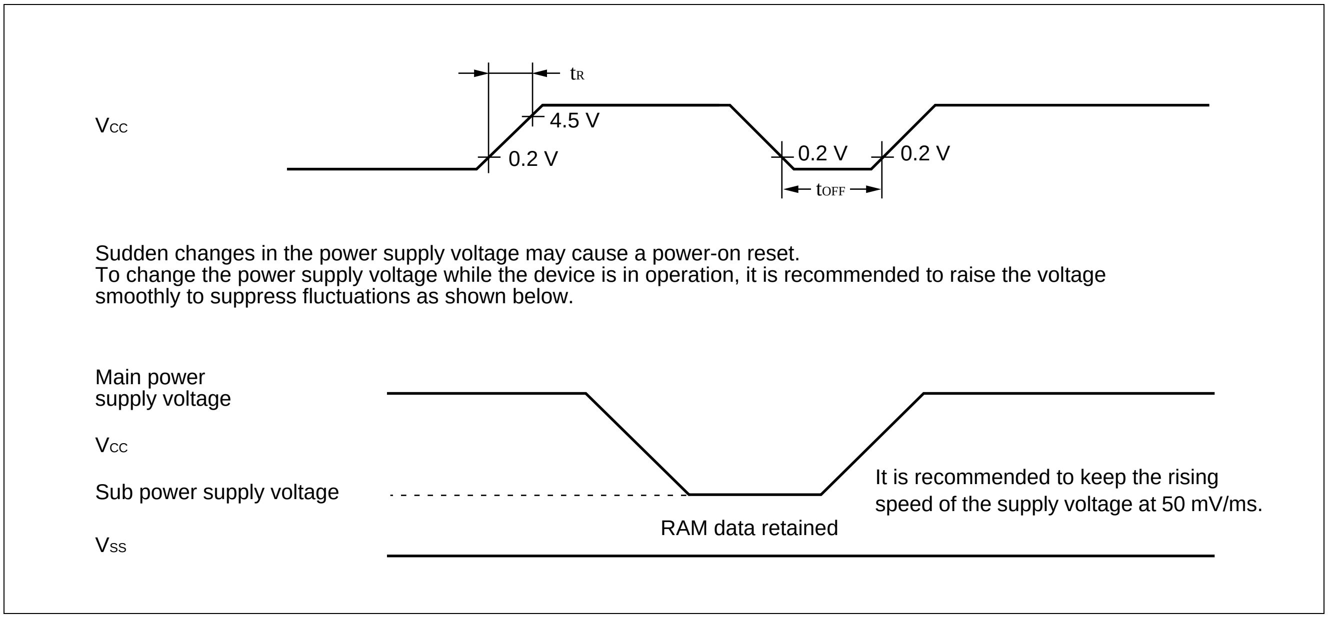
(2) Specification for Power-on Reset

($AV_{SS} = V_{SS} = 0.0$ V, $T_A = –30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Power supply rising time	t_R	V_{CC}	—	—	30	ms	*
Power supply cut-off time	t_{OFF}	V_{CC}		1	—	ms	Due to repeated operations

* : V_{CC} must be kept lower than 0.2 V before power-on.

Notes: • The above ratings are values for causing a power-on reset.
- When $\overline{HST}$ is set to “L”, apply power according to this table to cause a power-on reset irrespective of whether or not a power-on reset is required.
- For built-in resources in the device, re-apply power to the resources to cause a power-on reset.



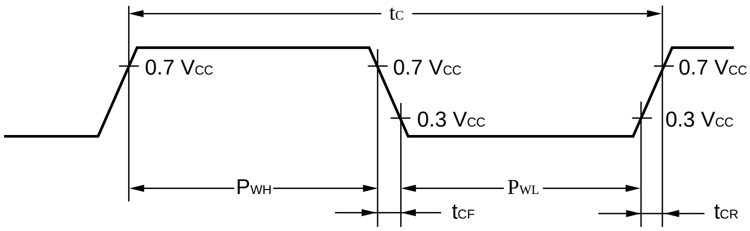
(3) Clock Timings

- **Operation at 5.0 V ±10%**

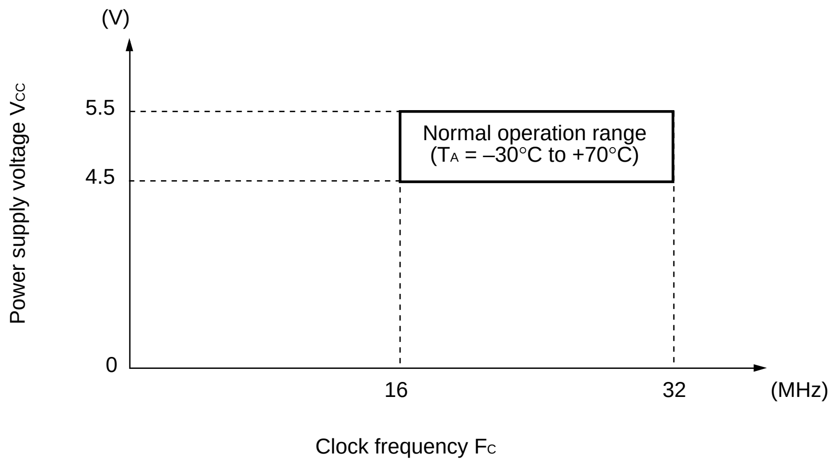
($AV_{SS} = V_{SS} = 0.0$ V, $T_A = -30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value			Unit	Remarks
				Min.	Typ.	Max.		
Clock frequency	F_C	X0, X1	V_{CC} = 5.0 V ±10%	16	—	32	MHz	
Clock cycle time	t_C	X0, X1	—	1/Fc	—	—	ns	
Input clock pulse width	P_{WH}, P_{WL}	X0	—	10	—	—	ns	Recommended duty ratio of 30% to 70%
Input clock rising/ falling time	t_{CR}, t_{CF}	X0	V_{CC} = 5.0 V ±10%	—	—	11	ns	Maximum value = t_{CR} + t_{CF}

- **Clock timings**



- **Relationship between clock frequency and power supply voltage**



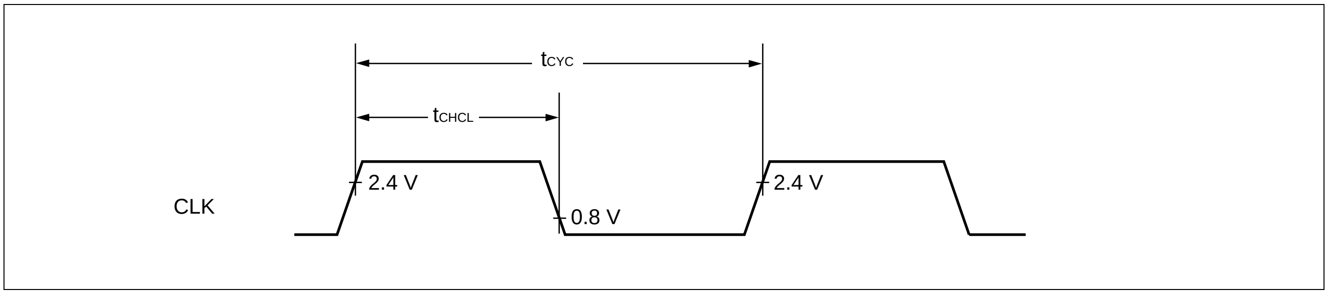
(4) Clock Output Timing

(AV_{CC} = V_{CC} = 4.5 V to 5.5 V, AV_{SS} = V_{SS} = 0.0 V, T_A = –30°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Cycle time (machine cycle)	t_{CYC}	CLK	—	2 t_C*1	32t_C*1*2	ns	
CLK ↑ → CLK ↓	t_{CHCL}	CLK	V_{CC} = 5.0 V ±10%	1 t_{CYC}/2 – 20	1 t_{CYC}/2 + 20	ns	

*1: For t_C (clock cycle time), refer to "(3) Clock Timings."

*2: This case is applied when the lowest speed (1/16) is selected by the clock gear function with the clock frequency (F_C) set at 16 MHz.



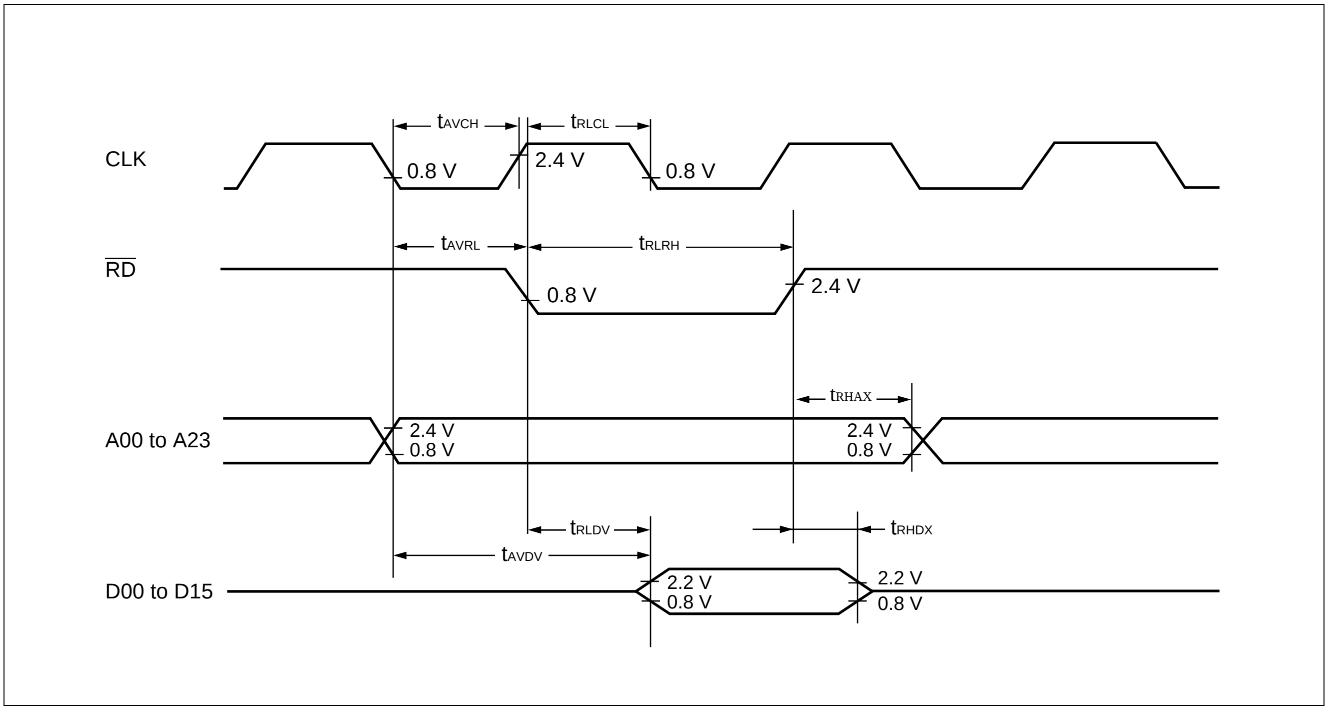
(3) Bus Read Timing

($AV_{CC} = V_{CC}$ = 2.7 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –40°C to +85°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Effective address → $\overline{RD}$ ↓ time	t_{AVRL}	A00 to A23	V_{CC} = 5.0 V ±10%	1 t_{CYC}*/2 – 20	—	ns	
Effective address → effective data input	t_{AVDV}	D15 to D00		—	(N + 1.5) × 1 t_{CYC}* – 40	ns	
$\overline{RD}$ pulse width	t_{RLRH}	$\overline{RD}$	—	(N + 1) × 1 t_{CYC}* – 25	—	ns	
$\overline{RD}$ ↓ → effective data input	t_{RLDV}	D15 to D00	V_{CC} = 5.0 V ±10%	—	(N + 1) × 1 t_{CYC}* – 30	ns	
$\overline{RD}$ ↑ → data hold time	t_{RHDX}	D15 to D00	—	0	—	ns	
$\overline{RD}$ ↑ → address effective time	t_{RHAX}	A00 to A23		1 t_{CYC}*/2 – 20	—	ns	
Effective address → CLK ↑ time	t_{AVCH}	CLK, A00 to A23		1 t_{CYC}*/2 – 25	—	ns	
$\overline{RD}$ ↓ → CLK ↑ time	t_{RLCL}	$\overline{RD}$, CLK		1 t_{CYC}*/2 – 25	—	ns	

N: Stands for the number of wait cycles. With no wait, N is set at “0”. (The number of wait cycles depends on an automatic wait and external RDY.)

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), “Clock output timing.”



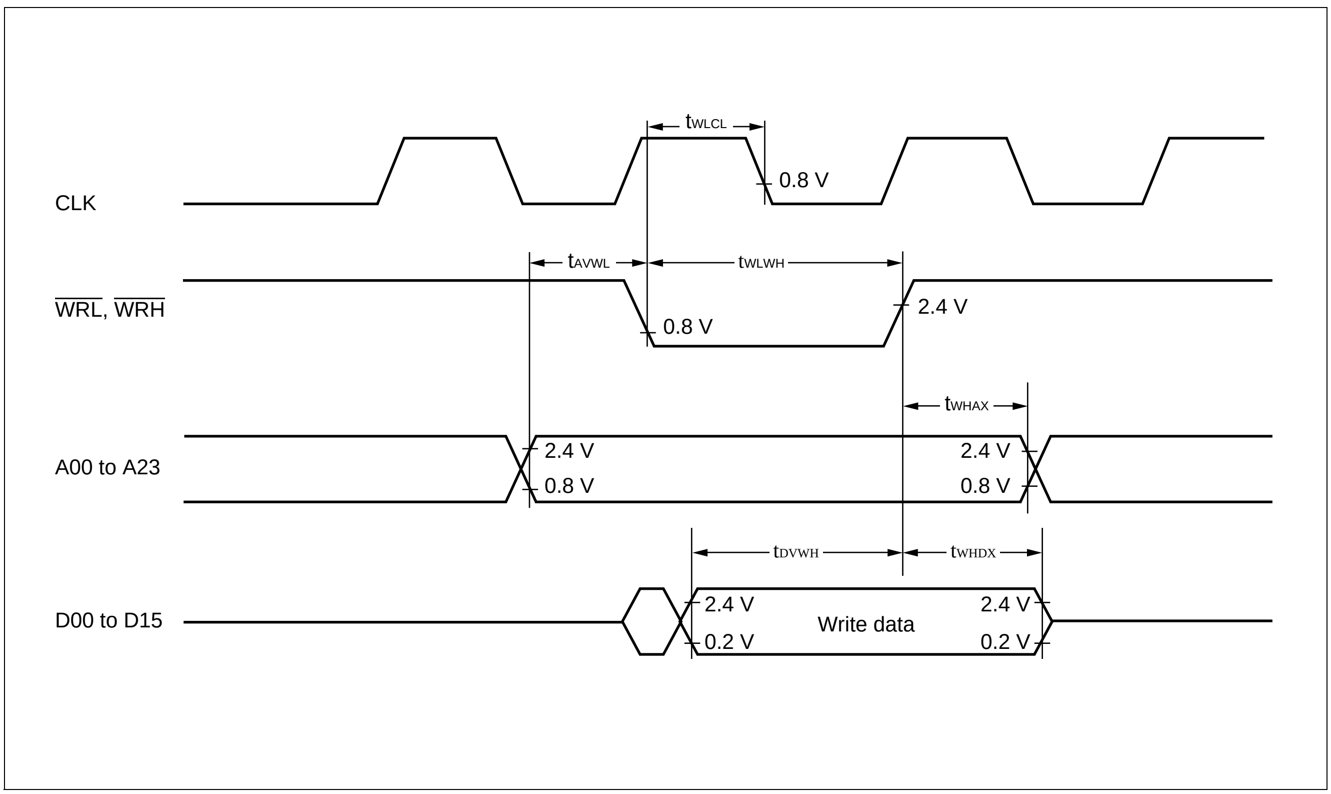
(4) Bus Write Timing

($AV_{CC} = V_{CC} = 4.5$ V to 5.5 V, $AV_{SS} = V_{SS} = 0.0$ V, $T_A = -30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Effective address → $\overline{\text{WRL}}$, $\overline{\text{WRH}}$ ↓ time	t_{AVWL}	A00 to A23	V_{CC} = 5.0 V ±10%	1 t_{CYC}*/2 – 20	—	ns	
$\overline{\text{WRL}}$, $\overline{\text{WRH}}$ pulse width	t_{WLWH}	$\overline{\text{WRL}}$, $\overline{\text{WRH}}$		(N + 1) × 1 t_{CYC}* – 25	—	ns	
Write data → $\overline{\text{WRL}}$, $\overline{\text{WRH}}$ ↑ time	t_{DVWH}	D15 to D00		(N + 1) × 1 t_{CYC}* – 40	—	ns	
$\overline{\text{WRL}}$, $\overline{\text{WRH}}$ ↑ → data hold time	t_{WHDX}	D15 to D00	V_{CC} = 5.0 V ±10%	1 t_{CYC}*/2 – 20	—	ns	
$\overline{\text{WRL}}$, $\overline{\text{WRH}}$ ↑ → address effective time	t_{WHAX}	A00 to A23	—	1 t_{CYC}*/2 – 20	—	ns	
$\overline{\text{WRL}}$, $\overline{\text{WRH}}$ ↓ → CLK ↓ time	t_{WLCL}	$\overline{\text{WRL}}$, CLK		1 t_{CYC}*/2 – 25	—	ns	

N: Stands for the number of wait cycles. With no wait, N is set at "0". (The number of wait cycles depends on an automatic wait and external RDY.)

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."



(5) Ready Input Timing

• **CLK signal standards**

(AV$_{CC}$ = V$_{CC}$ = 4.5 V to 5.5 V, AV$_{SS}$ = V$_{SS}$ = 0.0 V, T$_A$ = –30°C to +70°C)

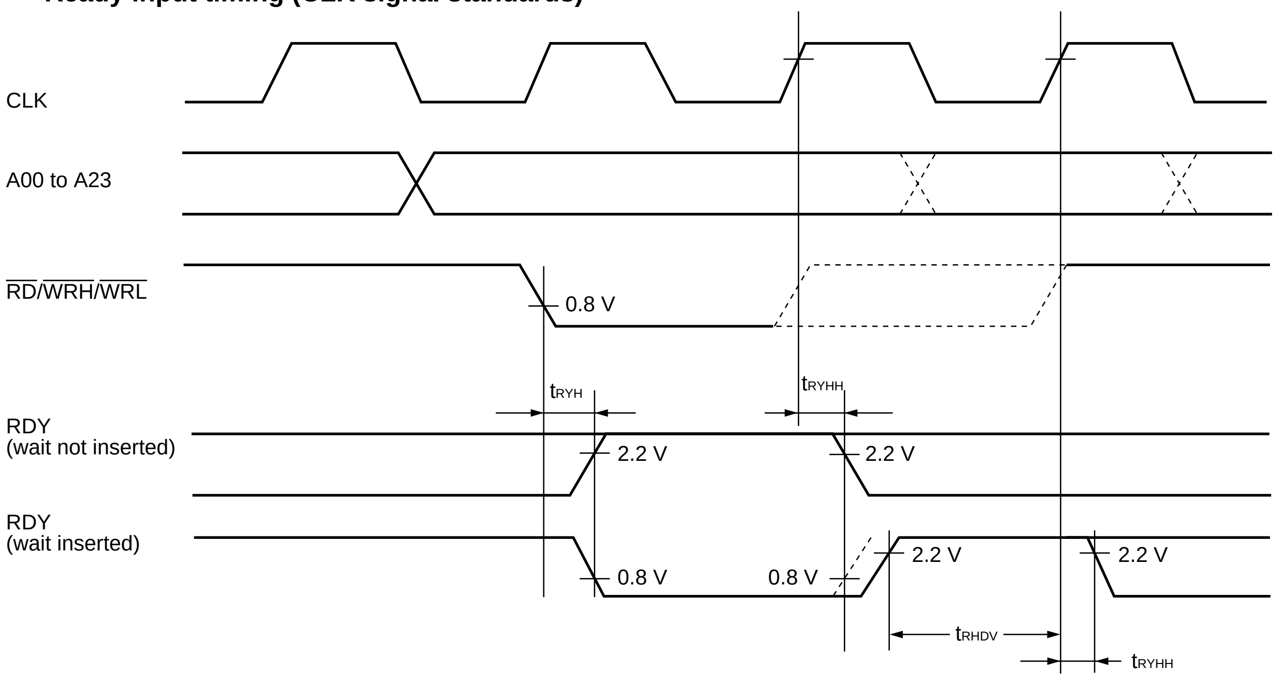
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
$\overline{RD}$/$\overline{WRH}$/$\overline{WRL}$ ↓ → RDY ↓ time	t_{RYHS}	$\overline{RD}$/$\overline{WRH}$/ $\overline{WRL}$, RDY		0	N ×1 t_{CYC}* + 15	ns	
RDY setup time (in diallocating)	t_{RHDV}	RDY	V$_{CC}$ = 5.0 V ±10%	30	—	ns	
RDY hold time	t_{RYHH}	RDY	—	0	—	ns	

N: Stands for the number of wait cycles. With no wait, N is set at "0". (The number of wait cycles depends on an automatic wait and external RDY.)

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."

Note: Use the automatic ready function when the setup time for the rising edge of the RDY signal is not sufficient.

• **Ready input timing (CLK signal standards)**



- **$\overline{\text{RD}}$/$\overline{\text{WRH}}$/$\overline{\text{WRL}}$ signal standards**

(AV$_{CC}$ = V$_{CC}$ = 4.5 V to 5.5 V, AV$_{SS}$ = V$_{SS}$ = 0.0 V, T$_A$ = –30°C to +70°C)

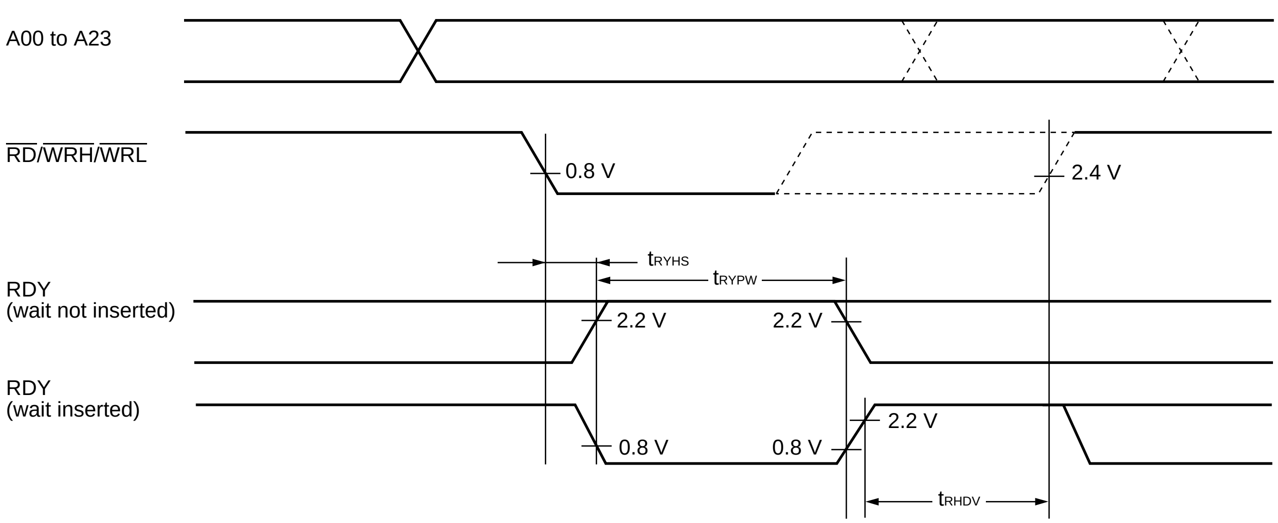
Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
$\overline{\text{RD}}$/$\overline{\text{WRH}}$/$\overline{\text{WRL}}$ ↓ → RDY ↓ time	t_{RYHS}	$\overline{\text{RD}}$/$\overline{\text{WRH}}$/ $\overline{\text{WRL}}$, RDY	—	0	N × 1 t_{CYC}*3 + 15*1	ns	
RDY pulse width	t_{RYPW}	RDY	V$_{CC}$ = 5.0 V ±10%	1/2 t_{CYC}*3 + 20	(m + 1) × 1 t_{CYC}*2,*3	ns	
RDY ↑ → $\overline{\text{RD}}$ ↑	t_{RHDV}	$\overline{\text{RD}}$/$\overline{\text{WRH}}$/ $\overline{\text{WRL}}$, RDY	—	1 t_{CYC}*3 – 15	2 t_{CYC}*3 – 25	ns	

N: Stands for the number of wait cycles. With no wait, N is set at “0”. (The number of wait cycles depends on an automatic wait and external RDY.)

m: Stands for the number of RDY wait cycles. With no wait, m is set at “0”.

*1: Use the automatic ready function when the setup time is not sufficient.
*2: If the pulse width has exceeded the maximum value, the wait period may be extended beyond the specified number of cycles by one cycle.
*3: For t_{CYC} (cycle time (machine cycle)), see paragraph (4), “Clock output timing.”

- **Ready input timing ($\overline{\text{RD}}$/$\overline{\text{WRH}}$/$\overline{\text{WRL}}$ signal standards)**



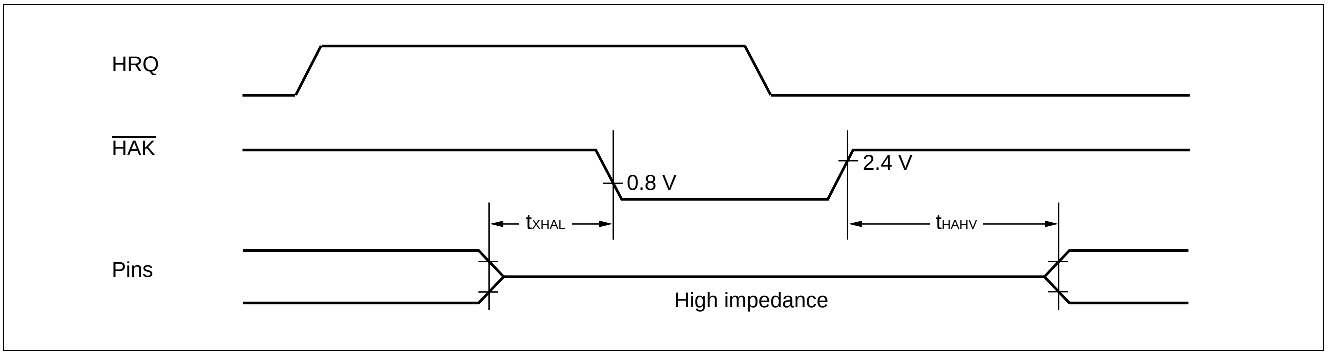
(8) Hold Timing

($AV_{CC} = V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –30°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Pins in floating status → $\overline{HAK}$ ↓ time	t_{XHAL}	$\overline{HAK}$	V_{CC} = 5.0 V ±10%	30	1 t_{CYC}*	ns	
$\overline{HAK}$ ↑ → pin valid time	t_{HAHV}	$\overline{HAK}$	—	1 t_{CYC}*	2 t_{CYC}*	ns	

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), “Clock output timing.”

Note: More than 1 machine cycle is needed before $\overline{HAK}$ changes after HRQ pin is fetched.



(9) UART Timing

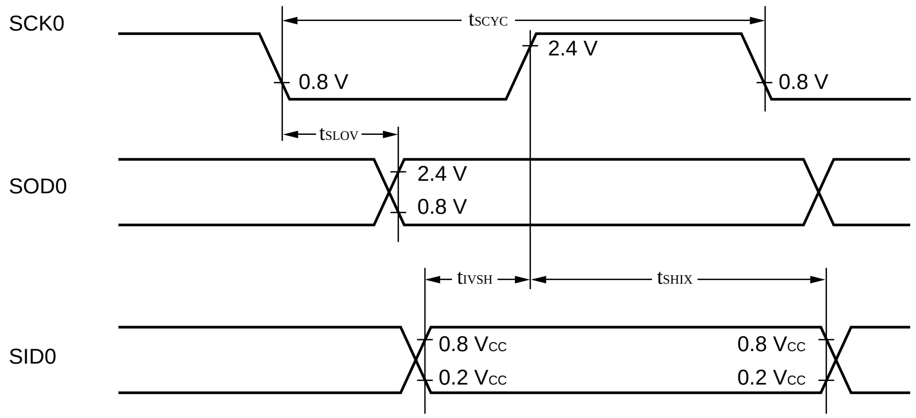
($AV_{CC} = V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, T_A = –30°C to +70°C)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	SCK0	—	8 t_{CYC}*	—	ns	Internal shift clock mode C_L = 80 pF for an output pin
SCK ↓ → SOD delay time	t_{SLOV}	SCK0, SOD0	V_{CC} = 5.0 V ±10%	–80	80	ns	
Valid SID → SCK ↑	t_{IVSH}	SCK0, SID0		100	—	ns	
SCK ↑ → valid SID hold time	t_{SHIX}	SCK0, SID0		60	—	ns	
Serial clock “H” pulse width	t_{SHSL}	SCK0	—	4 t_{CYC}*	—	ns	External shift clock mode C_L = 80 pF for an output pin
Serial clock “L” pulse width	t_{SLSH}	SCK0		4 t_{CYC}*	—	ns	
SCK ↓ → SOD delay time	t_{SLOV}	SCK0, SID0	V_{CC} = 5.0 V ±10%	—	150	ns	
Valid SID → SCK ↑	t_{IVSH}	—		60	—	ns	
SCK ↑ → valid SID hold time	t_{SHIX}	SCK0, SID0		60	—	ns	

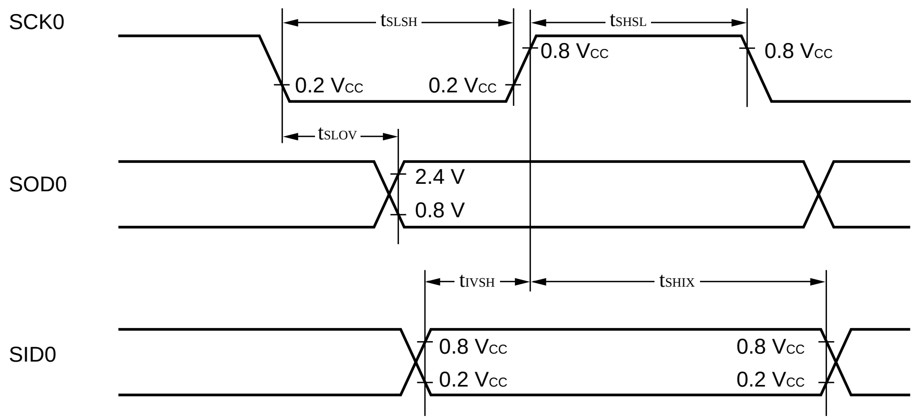
* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), “Clock output timing.”

Notes: • These are AC ratings in the CLK synchronous mode.
• C_L is the load capacitor value connected to pins while testing.

- **Internal shift clock mode**



- **External shift clock mode**

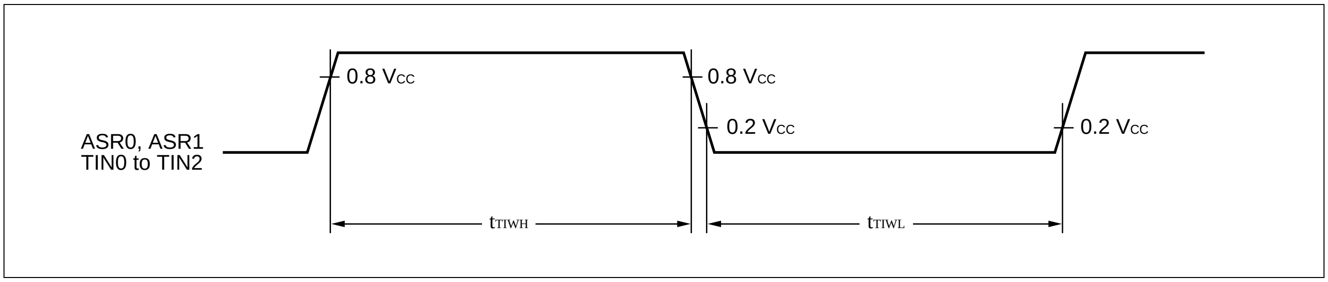


(10) Timer Input Timing

($AV_{CC} = V_{CC} = 4.5$ V to 5.5 V, $AV_{SS} = V_{SS} = 0.0$ V, $T_A = –30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	t_{TIWH}, t_{TIWL}	ASR0, ASR1, TIN0 to TIN2	—	4 t_{CYC}*	—	ns	

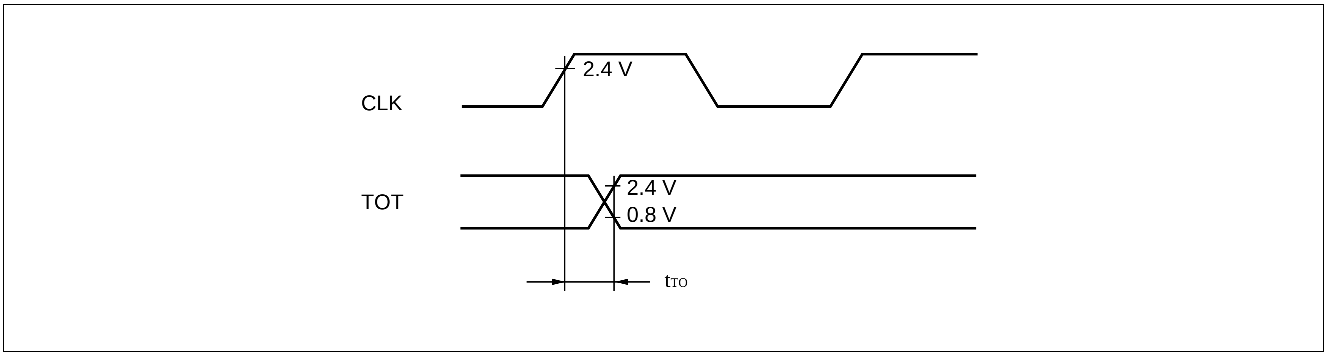
* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."



(11) Timer Output Timing

($AV_{CC} = V_{CC} = 4.5$ V to 5.5 V, $AV_{SS} = V_{SS} = 0.0$ V, $T_A = –30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
CLK ↑ → TOT transition time	t_{TO}	TOT0 to TOT2, PWM0 to PWM3	$V_{CC} = 5.0$ V ±10%	—	40	ns	



(12) I/O Simple Serial Timing

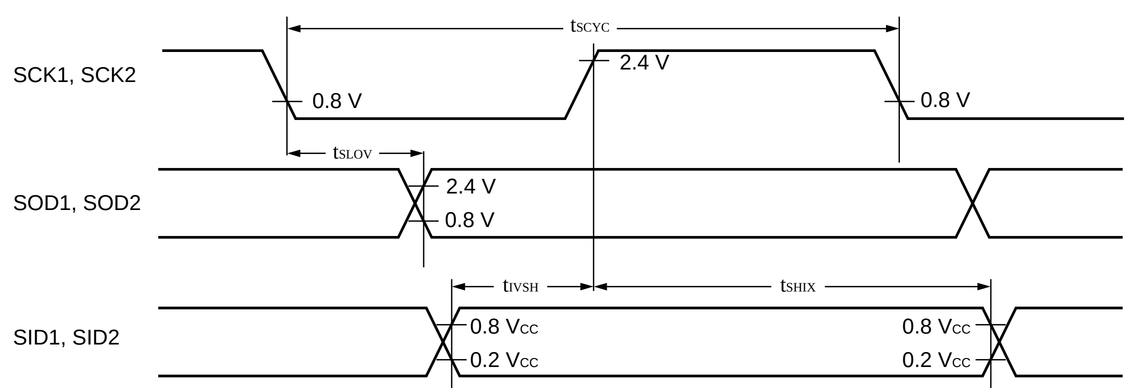
($AV_{CC} = V_{CC} = 4.5$ V to 5.5 V, $AV_{SS} = V_{SS} = 0.0$ V, $T_A = -30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Serial clock cycle time	t_{SCYC}	SCK1, SCK2	—	2 t_{CYC}*	—	ns	Internal shift clock mode C_L = 80 pF for an output pin
SCK ↓ → SOD delay time	t_{SLOV}	SCK1, SOD1, SCK2, SOD2,		—	1 t_{CYC}*/2	ns	
Valid SID → SCK ↑	t_{IVSH}	SCK1, SID1, SCK2, SID2,		1 t_{CYC}*	—	ns	
SCK ↑ → valid SID hold time	t_{SHIX}	SCK1, SID1, SCK2, SID2,		1 t_{CYC}*	—	ns	

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."

Note: C_L is the load capacitor value connected to pins while testing.

- **Internal shift clock mode**

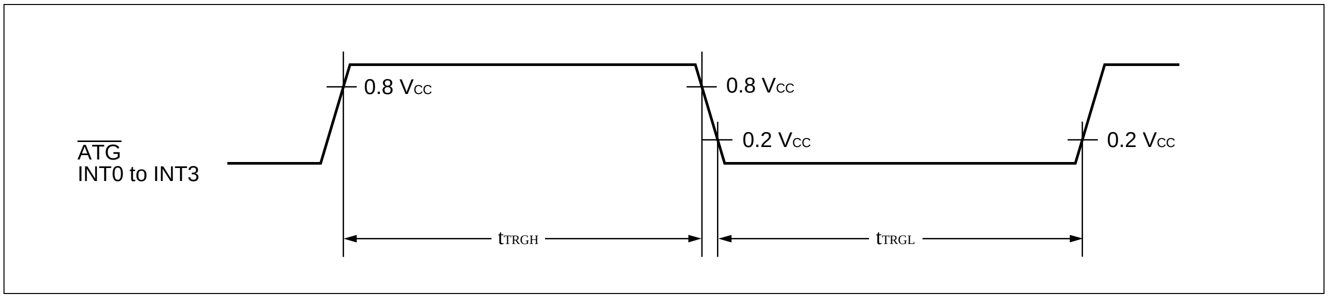


(13) Trigger input timing

($AV_{CC} = V_{CC} = 4.5$ V to 5.5 V, $AV_{SS} = V_{SS} = 0.0$ V, $T_A = -30°C$ to $+70°C$)

Parameter	Symbol	Pin name	Condition	Value		Unit	Remarks
				Min.	Max.		
Input pulse width	t_{TRGH}, t_{TRGL}	$\overline{ATG}$, INT0 to INT3	—	5 t_{CYC}*	—	ns	

* : For t_{CYC} (cycle time (machine cycle)), see paragraph (4), "Clock output timing."

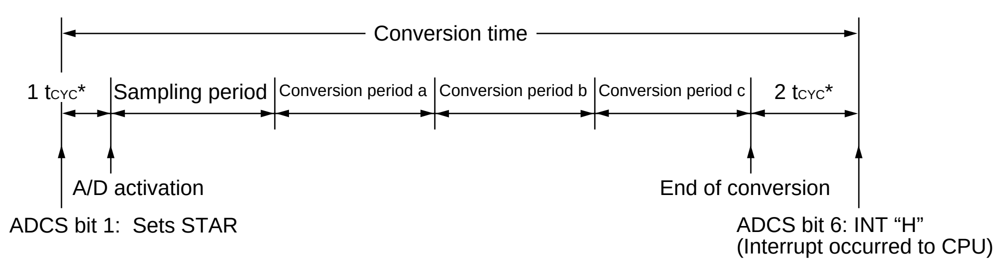


5. A/D Converter Electrical Characteristics

(AV_{CC} = V_{CC} = 4.5 V to 5.5 V, AV_{SS} = V_{SS} = 0.0 V, T_A = –30°C to +70°C)

Parameter		Symbol	Pin name	Condition	Value			Unit
					Min.	Typ.	Max.	
Resolution		—	—	—	—	8, 10	10	bit
Total error		—	—		—	—	±3.0	LSB
Linearity error		—	—		—	—	±2.0	LSB
Differential linearity error		—	—		—	—	±1.9	LSB
Zero transition voltage		V_{OT}	AN0 to AN7		AVRL – 1.0 LSB	AVRL + 1.0 LSB	AVRL + 3.0 LSB	mV
Full-scale transition voltage		V_{FST}	AN0 to AN7		AVRH – 4.0 LSB	AVRH – 1.0 LSB	AVRH + 1.0 LSB	mV
Conversion time*1		—	—	Use the A/D data register for setup. V_{CC} = 5.0 V ±10%	1.25	—	—	µs
	Sampling period	—	—		560	—	—	ns
	Conversion period a	—	—		125	—	—	ns
	Conversion period b	—	—		125	—	—	ns
	Conversion period c	—	—		250	—	—	ns
Analog port input current		I_{AIN}	AN0 to AN7	—	—	0.1	3	µA
Analog input voltage		V_{AIN}	AN0 to AN7		AVRL	—	AVRH	V
Reference voltage		—	AVRH	AVRH – AVRL ≧ 2.7	AVRL + 2.7	—	AV_{CC}	V
		—	AVRL		0	—	AVRH – 2.7	V
Power supply current		I_A	AV_{CC}	—	—	15	20	mA
		I_{AS}*2	AV_{CC}	Supply current when the CPU stops (AV_{CC} = 5.5 V)	—	—	5	µA
Reference voltage supply current		I_R	AVRH	—	—	0.7	2	µA
		I_{RS}*2	AVRH	Supply current when the CPU stops (AV_{CC} = 5.5 V)	—	—	5	µA
Offset between channels		—	AN0 to AN7	—	—	—	4	LSB

*1: Glossary for conversion time



* : For t_{CYC}, see ■ Electrical Characteristics, 4, "AC Characteristics," Cycle time (machine cycle) in paragraph (4), "Clock output timing."

*2: IAS and IRS signify currents when the A/D converter does not operate and when the CPU is out of service, respectively.

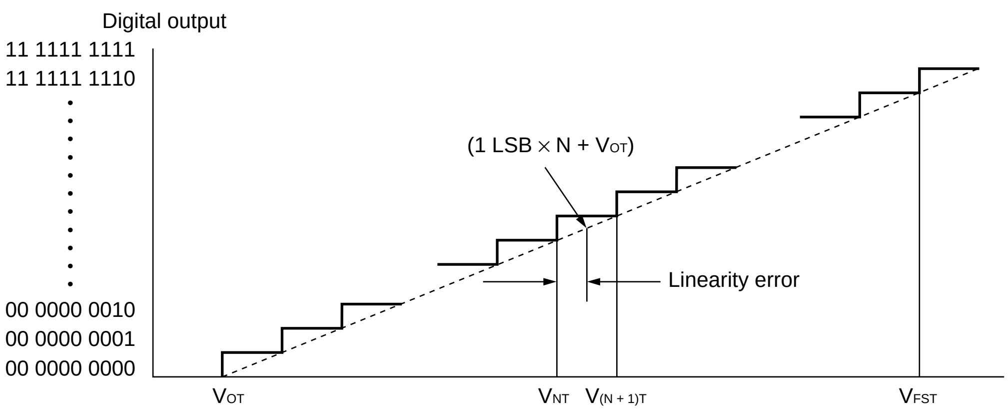
6. A/D Converter Glossary

Resolution: Analog changes that are identifiable with the A/D converter
With 10 bits supported, an analog voltage can be divided into 2^{10} parts.

Linearity error: The deviation of the straight line connecting the zero transition point (“00 0000 0000” ↔ “00 0000 0001”) with the full-scale transition point (“11 1111 1110” ↔ “11 1111 1111”) from actual conversion characteristics

Differential linearity error: The deviation of input voltage needed to change the output code by 1 LSB from the theoretical value

Total error: The total error is defined as a difference between the actual value and the theoretical value, which includes zero-transition error/full-scale transition error, linearity error, differential linearity error and error caused by noise.



$$1\ \text{LSB} = \frac{V_{FST} - V_{OT}}{1022}$$

$$\text{Linearity error} = \frac{V_{NT} - (1\ \text{LSB} \times N + V_{OT})}{1\ \text{LSB}}\ [\text{LSB}]$$

$$\text{Differential linearity error} = \frac{V_{(N+1)T} - V_{NT}}{1\ \text{LSB}} - 1\ \text{LSB}\ [\text{LSB}]$$

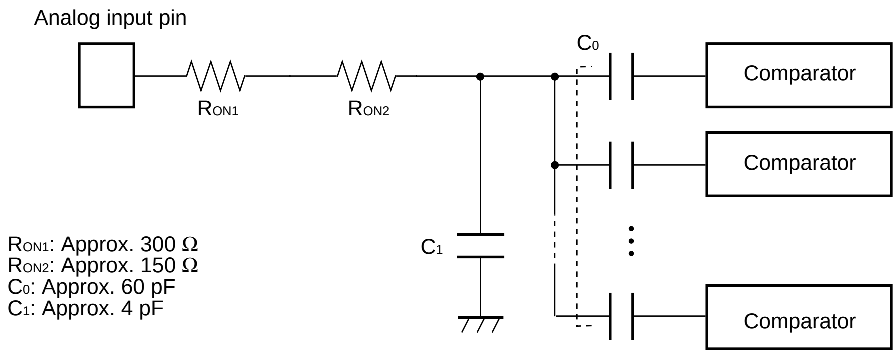
7. Notes on Using A/D Converter

Select the output impedance value for the external circuit of analog input according to the following conditions. Output impedance values of the external circuit of 300 Ω or lower are recommended.

When capacitors are connected to external pins, the capacitance of several thousand times the internal capacitor value is recommended to minimized the effect of voltage distribution between the external capacitor and internal capacitor.

When the output impedance of the external circuit is too high, the sampling time for analog voltages may not be sufficient (sampling time = 0.56 µs @machine clock of 16 MHz).

- **Block diagram of analog input circuit model**



Note: Listed values must be considered as standards.

- **Error**

The smaller the | AVRH – AVRL |, the greater the error would become relatively.

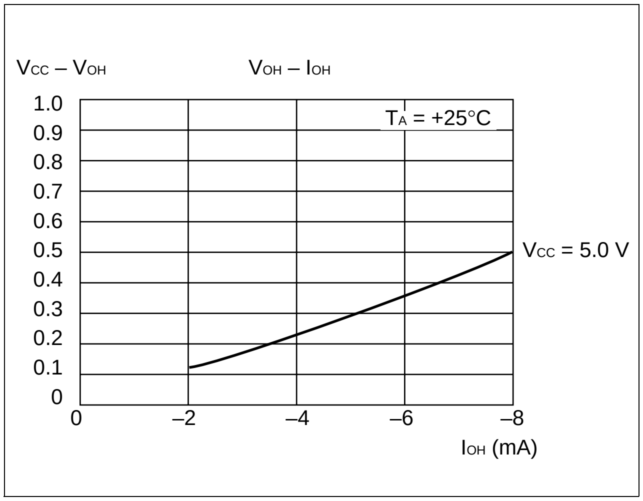
8. 8-bit D/A Converter Electrical Characteristics

(AV_{CC} = V_{CC} = 4.5 V to 5.5 V, AV_{SS} = V_{SS} = 0.0 V, T_A = –30°C to +70°C)

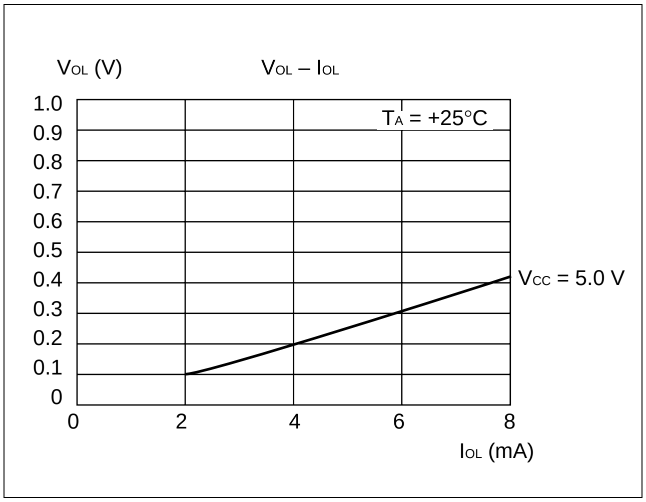
Parameter	Symbol	Pin name	Condition	Value			Unit
				Min.	Typ.	Max.	
Resolution	—	—	—	—	8	8	bit
Differential linearity error	—	—		—	—	±0.9	LSB
Absolute accuracy	—	—	V_{CC} = DVRH = 5.0 V, DVRL = 0.0 V	—	—	1.2	%
Conversion time	—	—	Load capacitance: 20 pF	—	10	20	µs
Analog power supply voltage	—	DVRH		V_{SS} + 2.0	—	V_{CC}	V
	—	DVRL	DVRH – DVRL ≧ 2.0 V	V_{SS}	—	V_{CC} – 2.0	V
Reference voltage supply current	I_D	DVRH	During conversion	—	1.0	1.5	mA
	I_{DH}	DVRH	When the CPU is stopped	—	—	10	µA
Analog output impedance	—	—	—	—	28	—	kΩ

■ EXAMPLE CHARACTERISTICS

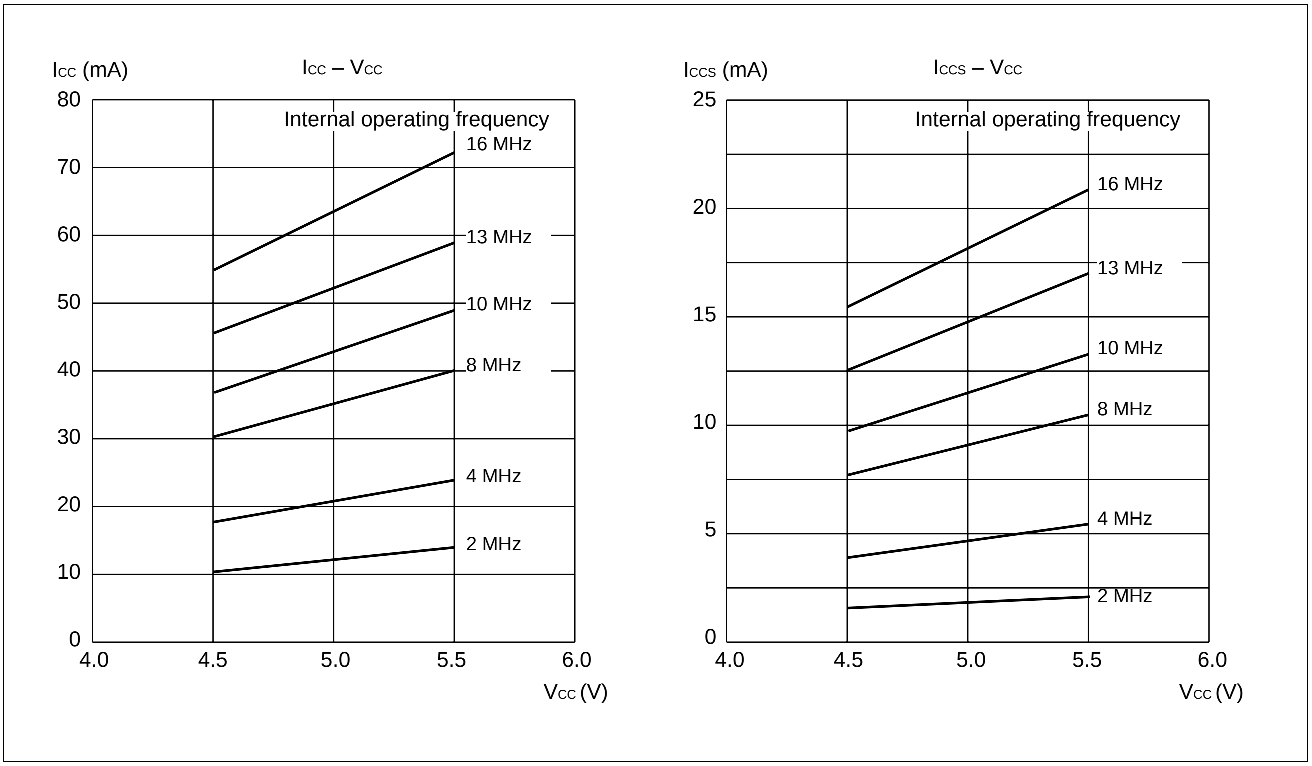
(1) "H" Level Output Voltage



(2) "L" Level Output Voltage



(3) Power Supply Current



■ INSTRUCTIONS (421 INSTRUCTIONS)

Table 1 Description of Items in Instruction List

Item	Description
Mnemonic	English upper case and symbol: Described directly in assembler code. English lower case: Converted in assembler code. Number of letters after English lower case: Describes bit width in code.
#	Describes number of bytes.
~	Describes number of cycles. For other letters in other items, refer to table 4.
B	Describes correction value for calculating number of actual states. Number of actual states is calculated by adding value in the ~section.
Operation	Describes operation of instructions.
LH	Describes a special operation to 15 bits to 08 bits of the accumulator. Z : Transfer 0. X : Sign-extend and transfer. – : No transmission
AH	Describes a special operation to the upper 16-bit of the accumulator. * : Transmit from AL to AH. – : No transfer. Z : Transfer 00_H to AH. X : Sign-extend AL and transfer 00_H or FF_H to AH.
I	Describes status of I (interrupt enable), S (stack), T (sticky bit), N (negative), Z (zero), V (overflow), and C (carry) flags. * : Changes after execution of instruction. – : No changes. S : Set after execution of instruction. R : Reset after execution of instruction.
S	
T	
N	
Z	
V	
C	
RMW	Describes whether or not the instruction is a read-modify-write type (a data is read out from memory etc. in single cycle, and the result is written into memory etc.). * : Read-modify-write instruction – : Not read-modify-write instruction Note: Not used to addresses having different functions for reading and writing operations.

Table 2 Description of Symbols in Instruction Table

Item	Description
A	32-bit accumlator The bit length is dependent on the instructions to be used. Byte : Lower 8-bit of AL Word :16-bit of AL Long : AL: 32-bit of AH
AH	Upper 16-bit of A
AL	Lower 16-bit of A
SP	Stack pointer (USP or SSP)
PC	Program counter
SPCU	Stack pointer upper limited register
SPCL	Stack pointer lower limited register
PCB	Program bank register
DTB	Data bank register
ADB	Additional data bank register
SSB	System stack bank register
USB	User stack bank register
SPB	Current stack bank register (SSB or USB)
DPR	Direct page register
brg1	DTB, ADB, SSB, USB, DPR, PCB
brg2	DTB, ADB, SSB, USB, DPR
Ri	R0, R1, R2, R3, R4, R5, R6, R7
RWi	RW0, RW1, RW2, RW3, RW4, RW5, RW6, RW7
RWj	RW0, RW1, RW2, RW3
RLi	RL0, RL1, RL2, RL3
dir addr16 addr24 ad24 0 to 15 ad24 16 to 23	Specify shortened direct address. Specify direct address. Specify physical direct address. bit0 to bit15 of addr24 bit16 to bit 23 of addr24
io	I/O area (000000_H to $0000FF_H$)
#imm4 #imm8 #imm16 #imm32 ext (imm8)	4-bit immediate data 8-bit immediate data 16-bit immediate data 32-bit immediate data 16-bit data calculated by sign-extending an 8-bit immediate data
disp8 disp16	8-bit displacement 16-bit displacement
bp	Bit offset value
vct4 vct8	Vector number (0 to 15) Vector number (0 to 255)

(Continued)

(Continued)

Item	Description
( )b	Bit address
rel ear eam	Specify PC relative branch. Specify effective address (code 00 to 07). Specify effective address (code 08 to 1F).
rlst	Register allocation

Table 3 Effective Address Field

Code	Symbol			Address type	Number of bytes in address extension block*
00 01 02 03 04 05 06 07	R0 R1 R2 R3 R4 R5 R6 R7	RW0 RW1 RW2 RW3 RW4 RW5 RW6 RW7	RL0 (RL0) RL1 (RL1) RL2 (RL2) RL3 (RL3)	Register direct "ea" corresponds to byte, word, and long word from left respectively.	—
08 09 0A 0B	@RW0 @RW1 @RW2 @RW3			Register indirect	0
0C 0D 0E 0F	@RW0 + @RW1 + @RW2 + @RW3 +			Register indirect with post increment	0
10 11 12 13 14 15 16 17	@RW0 + disp8 @RW1 + disp8 @RW2 + disp8 @RW3 + disp8 @RW4 + disp8 @RW5 + disp8 @RW6 + disp8 @RW7 + disp8			Register indirect with 8-bit displacement	1
18 19 1A 1B	@RW0 + disp16 @RW1 + disp16 @RW2 + disp16 @RW3 + disp16			Register indirect with 16-bit displacement	2
1C 1D 1E 1F	@RW0 + RW7 @RW1 + RW7 @PC + disp16 addr16			Register indirect with index Register indirect with index PC indirect with 16-bit displacement Direct address	0 0 2 2

Note: Number of bytes for address extension corresponds to “+” in the # (number of bytes) part in the instruction table.

Table 4 Number of Execution Cycles in Addressing Modes

Code	Operand	(a)* Number of execution cycles for addressing modes
00 to 07	Ri RWi RLi	Listed in instruction table
08 to 0B	@RWj	1
0C to 0F	@RWj +	4
10 to 17	@RWi + disp8	1
18 to 1B	@RWj + disp16	1
1C 1D 1E 1F	@RW0 + RW7 @RW1 + RW7 @PC + disp16 addr16	2 2 2 1

Note: (a) is used for ~ (number of cycles) and B (correction value) in instruction table.

Table 5 Correction Value for Number of Cycles for Calculating Actual Number of Cycles

Operand	(b)*	(c)*	(d)*
	byte	word	long
Internal register	+0	+0	+0
Internal RAM even address Internal RAM odd address	+0 +0	+0 +1	+0 +2
Other than internal RAM even address Other than internal RAM odd address	+1 +1	+1 +3	+2 +6
External data bus 8-bit	+1	+3	+6

Notes: • (b), (c), (d) is used for ~ (number of cycles) and B (correction value) in instruction table.

Table 6 Transmission Instruction (Byte) [50 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOV A, dir	2	2	(b)	byte (A) ← (dir)	Z	*	–	–	–	*	*	–	–	–
MOV A, addr16	3	2	(b)	byte (A) ← (addr16)	Z	*	–	–	–	*	*	–	–	–
MOV A, Ri	1	1	0	byte (A) ← (Ri)	Z	*	–	–	–	*	*	–	–	–
MOV A, ear	2	1	0	byte (A) ← (ear)	Z	*	–	–	–	*	*	–	–	–
MOV A, eam	2 +	2 + (a)	(b)	byte (A) ← (eam)	Z	*	–	–	–	*	*	–	–	–
MOV A, io	2	2	(b)	byte (A) ← (io)	Z	*	–	–	–	*	*	–	–	–
MOV A, #imm8	2	2	0	byte (A) ← imm8	Z	*	–	–	–	*	*	–	–	–
MOV A, @A	2	2	(b)	byte (A) ← ((A))	Z	–	–	–	–	*	*	–	–	–
MOV A, @RLi + disp8	3	6	(b)	byte (A) ← ((RLi) + disp8)	Z	*	–	–	–	*	*	–	–	–
MOV A, @SP + disp8	3	3	(b)	byte (A) ← ((SP) + disp8)	Z	*	–	–	–	*	*	–	–	–
MOVP A, addr24	5	3	(b)	byte (A) ← (addr24)	Z	*	–	–	–	*	*	–	–	–
MOVP A, @A	2	2	(b)	byte (A) ← ((A))	Z	–	–	–	–	*	*	–	–	–
MOVN A, #imm4	1	1	0	byte (A) ← imm4	Z	*	–	–	–	R	*	–	–	–
MOVX A, dir	2	2	(b)	byte (A) ← (dir)	X	*	–	–	–	*	*	–	–	–
MOVX A, addr16	3	2	(b)	byte (A) ← (addr16)	X	*	–	–	–	*	*	–	–	–
MOVX A, Ri	2	1	0	byte (A) ← (Ri)	X	*	–	–	–	*	*	–	–	–
MOVX A, ear	2	1	0	byte (A) ← (ear)	X	*	–	–	–	*	*	–	–	–
MOVX A, eam	2 +	2 + (a)	(b)	byte (A) ← (eam)	X	*	–	–	–	*	*	–	–	–
MOVX A, io	2	2	(b)	byte (A) ← (io)	X	*	–	–	–	*	*	–	–	–
MOVX A, #imm8	2	2	0	byte (A) ← imm8	X	*	–	–	–	*	*	–	–	–
MOVX A, @A	2	2	(b)	byte (A) ← ((A))	X	–	–	–	–	*	*	–	–	–
MOVX A, @RWi + disp8	2	3	(b)	byte (A) ← ((RWi) + disp8)	X	*	–	–	–	*	*	–	–	–
MOVX A, @RLi + disp8	3	6	(b)	byte (A) ← ((RLi) + disp8)	X	*	–	–	–	*	*	–	–	–
MOVX A, @SP + disp8	3	3	(b)	byte (A) ← ((SP) + disp8)	X	*	–	–	–	*	*	–	–	–
MOVPX A, addr24	5	3	(b)	byte (A) ← (addr24)	X	*	–	–	–	*	*	–	–	–
MOVPX A, @A	2	2	(b)	byte (A) ← ((A))	X	–	–	–	–	*	*	–	–	–
MOV dir, A	2	2	(b)	byte (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV addr16, A	3	2	(b)	byte (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, A	1	1	0	byte (Ri) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV ear, A	2	2	0	byte (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV eam, A	2 +	2 + (a)	(b)	byte (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV io, A	2	2	(b)	byte (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV @RLi + disp8, A	3	6	(b)	byte ((RLi) + disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV @SP + disp8, A	3	3	(b)	byte ((SP) + disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVP addr24, A	5	3	(b)	byte (addr24) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV Ri, ear	2	2	0	byte (Ri) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOV Ri, eam	2 +	3 + (a)	(b)	byte (Ri) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVP @A, Ri	2	3	(b)	byte ((A)) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV ear, Ri	2	3	0	byte (ear) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV eam, Ri	2 +	3 + (a)	(b)	byte (eam) ← (Ri)	–	–	–	–	–	*	*	–	–	–
MOV Ri, #imm8	2	2	0	byte (Ri) ← imm8	–	–	–	–	–	*	*	–	–	–
MOV io, #imm8	3	3	(b)	byte (io) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV dir, #imm8	3	3	(b)	byte (dir) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV ear, #imm8	3	2	0	byte (ear) ← imm8	–	–	–	–	–	*	*	–	–	–
MOV eam, #imm8	3 +	2 + (a)	(b)	byte (eam) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV @AL, AH	2	2	(b)	byte ((A)) ← (AH)	–	–	–	–	–	*	*	–	–	–
XCH A, ear	2	3	0	byte (A) ↔ (ear)	Z	–	–	–	–	–	–	–	–	–
XCH A, eam	2 +	3 + (a)	2 × (b)	byte (A) ↔ (eam)	Z	–	–	–	–	–	–	–	–	–
XCH Ri, ear	2	4	0	byte (Ri) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCH Ri, eam	2 +	5 + (a)	2 × (b)	byte (Ri) ↔ (eam)	–	–	–	–	–	–	–	–	–	–

Note: For (a) and (b), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 7 Transmission Instruction (Word) [40 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVW A, dir	2	2	(c)	word (A) ← (dir)	–	*	–	–	–	*	*	–	–	–
MOVW A, addr16	3	2	(c)	word (A) ← (addr16)	–	*	–	–	–	*	*	–	–	–
MOVW A, SP	1	2	0	word (A) ← (SP)	–	*	–	–	–	*	*	–	–	–
MOVW A, RWi	1	1	0	word (A) ← (RWi)	–	*	–	–	–	*	*	–	–	–
MOVW A, ear	2	1	0	word (A) ← (ear)	–	*	–	–	–	*	*	–	–	–
MOVW A, eam	2 +	2 + (a)	(c)	word (A) ← (eam)	–	*	–	–	–	*	*	–	–	–
MOVW A, io	2	2	(c)	word (A) ← (io)	–	*	–	–	–	*	*	–	–	–
MOVW A, @A	2	2	(c)	word (A) ← ((A))	–	–	–	–	–	*	*	–	–	–
MOVW A, #imm16	3	2	0	word (A) ← imm16	–	*	–	–	–	*	*	–	–	–
MOVW A, @RWi + disp8	2	3	(c)	word (A) ← ((RWi)	–	*	–	–	–	*	*	–	–	–
MOVW A, @RLi + disp8	3	6	(c)	+disp8)	–	*	–	–	–	*	*	–	–	–
MOVW A, @SP + disp8	3	3	(c)	word (A) ← ((RLi) +disp8)	–	*	–	–	–	*	*	–	–	–
MOVPW A, addr24	5	3	(c)	word (A) ← ((SP) + disp8)	–	*	–	–	–	*	*	–	–	–
MOVPW A, @A	2	2	(c)	word (A) ← (addr24)	–	–	–	–	–	*	*	–	–	–
				word (A) ← ((A))										
MOVW dir, A	2	2	(c)		–	–	–	–	–	*	*	–	–	–
MOVW addr16, A	3	2	(c)	word (dir) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW SP, #imm16	4	2	0	word (addr16) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW SP, A	1	2	0	word (SP) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW RWi, A	1	1	0	word (SP) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW ear, A	2	2	0	word (RWi) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW eam, A	2 +	2 + (a)	(c)	word (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW io, A	2	2	(c)	word (eam) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RWi + disp8, A	2	3	(c)	word (io) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW @RLi + disp8, A	3	6	(c)	word ((RWi) +disp8) ←	–	–	–	–	–	*	*	–	–	–
MOVW @SP + disp8, A	3	3	(c)	(A)	–	–	–	–	–	*	*	–	–	–
MOVPW addr24, A	5	3	(c)	word ((RLi) +disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVPW @A, RWi	2	3	(c)	word ((SP) + disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, ear	2	2	0	word (addr24) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, eam	2 +	3 + (a)	(c)	word ((A)) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW ear, RWi	2	3	0	word (RWi) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVW eam, RWi	2 +	3 + (a)	(c)	word (RWi) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVW RWi, #imm16	3	2	0	word (ear) ← (RWi)	–	–	–	–	–	*	*	–	–	–
MOVW io, #imm16	4	3	(c)	word (eam) ← (RWi)	–	–	–	–	–	–	–	–	–	–
MOVW ear, #imm16	4	2	0	word (RWi) ← imm16	–	–	–	–	–	*	*	–	–	–
MOVW eam, #imm16	4 +	2 + (a)	(c)	word (io) ← imm16	–	–	–	–	–	–	–	–	–	–
				word (ear) ← imm16										
MOVW @AL, AH	2	2	(c)	word (eam) ← imm16	–	–	–	–	–	*	*	–	–	–
XCHW A, ear	2	3	0	word ((A)) ← (AH)	–	–	–	–	–	–	–	–	–	–
XCHW A, eam	2 +	3 + (a)	2 × (c)		–	–	–	–	–	–	–	–	–	–
XCHW RWi, ear	2	4	0	word (A) ↔ (ear)	–	–	–	–	–	–	–	–	–	–
XCHW RWi, eam	2 +	5 + (a)	2 × (c)	word (A) ↔ (eam)	–	–	–	–	–	–	–	–	–	–
				word (RWi) ↔ (ear)										
				word (RWi) ↔ (eam)										

Note: For (a) and (c), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 8 Transmission Instruction (Long) [11 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVL A, ear	2	2	0	long (A) ← (ear)	–	–	–	–	–	*	*	–	–	–
MOVL A, eam	2 +	3 + (a)	(d)	long (A) ← (eam)	–	–	–	–	–	*	*	–	–	–
MOVL A, #imm32	5	3	0	long (A) ← imm32	–	–	–	–	–	*	*	–	–	–
MOVL A, @SP + disp8	3	4	(d)	long (A) ← ((SP) + disp8)	–	–	–	–	–	*	*	–	–	–
MOVPL A, addr24	5	4	(d)	long (A) ← (addr24)	–	–	–	–	–	*	*	–	–	–
MOVPL A, @A	2	3	(d)	long (A) ← ((A))	–	–	–	–	–	*	*	–	–	–
MOVPL @A, RLi	2	5	(d)	long ((A)) ← (RLi)	–	–	–	–	–	*	*	–	–	–
MOVL @SP + disp8, A	3	4	(d)	long ((SP) + disp8) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVPL addr24, A	5	4	(d)	long (addr24) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVL ear, A	2	2	0	long (ear) ← (A)	–	–	–	–	–	*	*	–	–	–
MOVL eam, A	2 +	3 + (a)	(d)	long (eam) ← (A)	–	–	–	–	–	*	*	–	–	–

Note: For (a) and (c), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 9 Add/Subtract (Byte, Word, Long) [42 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ADD A,#imm8	2	2	0	byte (A) ← (A) +imm8	Z	–	–	–	–	*	*	*	*	–
ADD A, dir	2	3	(b)	byte (A) ← (A) +(dir)	Z	–	–	–	–	*	*	*	*	–
ADD A, ear	2	2	0	byte (A) ← (A) +(ear)	Z	–	–	–	–	*	*	*	*	–
ADD A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) +(eam)	Z	–	–	–	–	*	*	*	*	–
ADD ear, A	2	2	0	byte (ear) ← (ear) + (A)	–	–	–	–	–	*	*	*	*	*
ADD eam, A	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) + (A)	Z	–	–	–	–	*	*	*	*	*
ADDC A	1	2	0	byte (A) ← (AH) + (AL) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, ear	2	2	0	byte (A) ← (A) + (ear) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDC A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) + (eam) + (C)	Z	–	–	–	–	*	*	*	*	–
ADDDC A	1	3	0	byte (A) ← (AH) + (AL) + (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
SUB A, #imm8	2	2	0	byte (A) ← (A) – imm8	Z	–	–	–	–	*	*	*	*	–
SUB A, dir	2	3	(b)	byte (A) ← (A) – (dir)	Z	–	–	–	–	*	*	*	*	–
SUB A, ear	2	2	0	byte (A) ← (A) – (ear)	Z	–	–	–	–	*	*	*	*	–
SUB A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) – (eam)	Z	–	–	–	–	*	*	*	*	–
SUB ear, A	2	2	0	byte (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	*
SUB eam, A	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBC A	1	2	0	byte (A) ← (AH) – (AL) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, ear	2	2	0	byte (A) ← (A) – (ear) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBC A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) – (eam) – (C)	Z	–	–	–	–	*	*	*	*	–
SUBDC A	1	3	0	byte (A) ← (AH) – (AL) – (C) (decimal)	Z	–	–	–	–	*	*	*	*	–
ADDW A	1	2	0	word (A) ← (AH) + (AL)	–	–	–	–	–	*	*	*	*	–
ADDW A, ear	2	2	0	word (A) ← (A) + (ear)	–	–	–	–	–	*	*	*	*	–
ADDW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) + (eam)	–	–	–	–	–	*	*	*	*	–
ADDW A, #imm16	3	2	0	word (A) ← (A) + imm16	–	–	–	–	–	*	*	*	*	–
ADDW ear, A	2	2	0	word (ear) – (ear) + (A)	–	–	–	–	–	*	*	*	*	*
ADDW eam, A	2 +	3 + (a)	2 × (c)	word (eam) – (eam) + (A)	–	–	–	–	–	*	*	*	*	*
ADDCW A, ear	2	2	0	word (A) ← (A) + (ear) + (C)	–	–	–	–	–	*	*	*	*	–
ADDCW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) + (eam) + (C)	–	–	–	–	–	*	*	*	*	–
SUBW A	1	2	0	word (A) ← (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
SUBW A, ear	2	2	0	word (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBW A, #imm16	3	2	0	word (A) ← (A) – imm16	–	–	–	–	–	*	*	*	*	–
SUBW ear, A	2	2	0	word (ear) ← (ear) – (A)	–	–	–	–	–	*	*	*	*	*
SUBW eam, A	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) – (A)	–	–	–	–	–	*	*	*	*	*
SUBCW A, ear	2	2	0	word (A) ← (A) – (ear) – (C)	–	–	–	–	–	*	*	*	*	–
SUBCW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) – (eam) – (C)	–	–	–	–	–	*	*	*	*	–
ADDL A, ear	2	5	0	long (A) ← (A) + (ear)	–	–	–	–	–	*	*	*	*	–
ADDL A, eam	2 +	6 + (a)	(d)	long (A) ← (A) + (eam)	–	–	–	–	–	*	*	*	*	–
ADDL A, #imm32	5	4	0	long (A) ← (A) + imm32	–	–	–	–	–	*	*	*	*	–
SUBL A, ear	2	5	0	long (A) ← (A) – (ear)	–	–	–	–	–	*	*	*	*	–
SUBL A, eam	2 +	6 + (a)	(d)	long (A) ← (A) – (eam)	–	–	–	–	–	*	*	*	*	–
SUBL A, #imm32	5	4	0	long (A) ← (A) – imm32	–	–	–	–	–	*	*	*	*	–

Note: For (a) to (d), refer to "Table 4 Number of Execution Cycles in Addressing Modes" and "Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles."

Table 10 Increment/Decrement (Byte, Word, Long) [12 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
INC ear	2	2	0	byte (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	*
INC eam	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DEC ear	2	2	0	byte (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	*
DEC eam	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*
INCW ear	2	2	0	word (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	*
INCW eam	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DECW ear	2	2	0	word (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	*
DECW eam	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*
INCL ear	2	4	0	long (ear) ← (ear) +1	–	–	–	–	–	*	*	*	–	–
INCL eam	2 +	5 + (a)	2 × (d)	long (eam) ← (eam) +1	–	–	–	–	–	*	*	*	–	*
DECL ear	2	4	0	long (ear) ← (ear) –1	–	–	–	–	–	*	*	*	–	*
DECL eam	2 +	5 + (a)	2 × (d)	long (eam) ← (eam) –1	–	–	–	–	–	*	*	*	–	*

Note: For (a) to (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 11 Compare (Byte, Word, Long) [11 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CMP A	1	1	0	byte (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMP A, ear	2	2	0	byte (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMP A, eam	2 +	3 + (a)	(b)	byte (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMP A, #imm8	2	2	0	byte (A) – imm8	–	–	–	–	–	*	*	*	*	–
CMPW A	1	1	0	word (AH) – (AL)	–	–	–	–	–	*	*	*	*	–
CMPW A, ear	2	2	0	word (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMPW A, eam	2 +	3 + (a)	(c)	word (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMPW A, #imm16	3	2	0	word (A) – imm16	–	–	–	–	–	*	*	*	*	–
CMPL A, ear	2	6	0	word (A) – (ear)	–	–	–	–	–	*	*	*	*	–
CMPL A, eam	2 +	7 + (a)	(d)	word (A) – (eam)	–	–	–	–	–	*	*	*	*	–
CMPL A, #imm32	5	3	0	word (A) – imm32	–	–	–	–	–	*	*	*	*	–

Note: For (a) to (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 12 Unsigned Multiply/Division (Word, Long) [11 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
DIVU A	1	*1	0	word (AH) /byte (AL) Quotient → byte (AL) Remainder → byte (AH)	–	–	–	–	–	–	–	*	*	–
DIVU A, ear	2	*2	0	word (A)/byte (ear) Quotient → byte (A) Remainder → byte (ear)	–	–	–	–	–	–	–	*	*	–
DIVU A, eam	2 +	*3	*6	word (A)/byte (eam) Quotient → byte (A) Remainder → byte (eam)	–	–	–	–	–	–	–	*	*	–
DIVUW A, ear	2	*4	0	long (A)/word (ear) Quotient → word (A) Remainder → word (ear)	–	–	–	–	–	–	–	*	*	–
DIVUW A, eam	2+	*5	*7	long (A)/word (eam) Quotient → word (A) Remainder → word (eam)	–	–	–	–	–	–	–	*	*	–
MULU A	1	*8	0	byte (AH) byte (AL) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU A, ear	2	*9	0	byte (A) byte (ear) → word (A)	–	–	–	–	–	–	–	–	–	–
MULU A, eam	2 +	*10	(b)	byte (A) byte (eam) → word (A)	–	–	–	–	–	–	–	–	–	–
MULUW A	1	*11	0	word (AH) word (AL) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW A, ear	2	*12	0	word (A) word (ear) → long (A)	–	–	–	–	–	–	–	–	–	–
MULUW A, eam	2 +	*13	(c)	word (A) word (eam) → long (A)	–	–	–	–	–	–	–	–	–	–

Note: For (b) and (c), refer to "Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles."

*1: Set to 3 when the division-by-0, 6 for an overflow, and 14 for normal operation.
*2: Set to 3 when the division-by-0, 6 for an overflow, and 13 for normal operation.
*3: Set to 5 + (a) when the division-by-0, 7 + (a) for an overflow, and 17 + (a) for normal operation.
*4: Set to 3 when the division-by-0, 5 for an overflow, and 21 for normal operation.
*5: Set to 4 + (a) when the division-by-0, 7 + (a) for an overflow, and 25 + (a) for normal operation.
*6: When the division-by-0, (b) for an overflow, and 2 × (b) for normal operation.
*7: When the division-by-0, (c) for an overflow, and 2 × (c) for normal operation.
*8: Set to 3 when byte (AH) is zero, 7 when byte (AH) is not zero.
*9: Set to 3 when byte (ear) is zero, 7 when byte (ear) is not zero.
*10:Set to 4 + (a) when byte (eam) is zero, 8 + (a) when byte (eam) is not zero.
*11:Set to 3 when word (AH) is zero, 11 when word (AH) is not zero.
*12:Set to 4 when word (ear) is zero, 11 when word (ear) is not zero.
*13:Set to 4 + (a) when word (eam) is zero, 12 + (a) when word (eam) is not zero.

Table 0 Signed multiplication/division (Word, Long) [11 Instructions]

Mnemonic		#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
DIV	A	2	*1	0	word (AH)/byte (AL) Quotient → byte (AL) Remainder → byte (AH)	Z	–	–	–	–	–	–	*	*	–
DIV	A, ear	2	*2	0	word (A)/byte (ear) Quotient → byte (A) Remainder → byte (ear)	Z	–	–	–	–	–	–	*	*	–
DIV	A, eam	2 +	*3	*6	word (A)/byte (eam) Quotient → byte (A) Remainder → byte (eam)	Z	–	–	–	–	–	–	*	*	–
DIVW	A, ear	2	*4	0	long (A)/word (ear) Quotient → word (A) Remainder → word (ear)	–	–	–	–	–	–	–	*	*	–
DIVW	A, eam	2 +	*5	*7	long (A)/word (eam) Quotient → word (A) Remainder → word (eam)	–	–	–	–	–	–	–	*	*	–
MUL	A	2	*8	0	byte (AH) × byte (AL) → word (A)	–	–	–	–	–	–	–	–	–	–
MUL	A, ear	2	*9	0	byte (A) × byte (ear) → word (A)	–	–	–	–	–	–	–	–	–	–
MUL	A, eam	2 +	*10	(b)	byte (A) × byte (eam) → word (A)	–	–	–	–	–	–	–	–	–	–
MULW	A	2	*11	0	word (AH) × word (AL) → long (A)	–	–	–	–	–	–	–	–	–	–
MULW	A, ear	2	*12	0	word (A) × word (ear) → long (A)	–	–	–	–	–	–	–	–	–	–
MULW	A, eam	2 +	*13	(b)	word (A) × word (eam) → long (A)	–	–	–	–	–	–	–	–	–	–

For (b) and (c), refer to “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: Set to 3 for divide-by-0, 8 or 18 for an overflow, and 18 for normal operation.
*2: Set to 3 for divide-by-0, 10 or 21 for an overflow, and 22 for normal operation.
*3: Set to 4 + (a) for divide-by-0, 11 + (a) or 22 + (a) for an overflow, and 23 + (a) for normal operation.
*4: Positive divided: Set to 4 for divide-by-0, 10 or 29 for an overflow, and 30 for normal operation.
Negative divided: Set to 4 for divide-by-0, 11 or 30 for an overflow, and 31 for normal operation.
*5: Positive divided: Set to 4 + (a) for divide-by-0, 11 + (a) or 30 + (a) for an overflow, and 31 + (a) for normal operation.
Negative divided: Set to 4 + (a) for divide-by-0, 12 + (a) or 31 + (a) for an overflow, and 32 + (a) for normal operation.
*6: Set to (b) when the division-by-0 or an overflow, and 2 × (b) for normal operation.
*7: Set to (c) when the division-by-0 or an overflow, and 2 × (c) for normal operation.
*8: Set to 3 when byte (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
*9: Set to 3 when byte (ear) is zero, 12 when the result is positive, and 13 when the result is negative.
*10: Set to 4 + (a) when byte (eam) is zero, 13 + (a) when the result is positive, and 14 + (a) when the result is negative.
*11: Set to 3 when word (AH) is zero, 12 when the result is positive, and 13 when the result is negative.
*12: Set to 3 when word (ear) is zero, 16 when the result is positive, and 19 when the result is negative.
*13: Set to 4 + (a) when word (eam) is zero, 17 + (a) when the result is positive, and 20 + (a) when the result is negative.

Note: When overflow occurs during DIV or DIVW instruction execution, the number of execution cycles takes two values because of detection before and after an operation.
When overflow occurs during DIV or DIVW instruction execution, the contents of AL are destroyed.

Table 14 Logic 1 (Byte, Word) [39 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
AND A, #imm8	2	2	0	byte (A) ← (A) and imm8	–	–	–	–	–	*	*	R	–	–
AND A, ear	2	2	0	byte (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
AND A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
AND ear, A	2	3	0	byte (ear) ← (ear) and (A)	–	–	–	–	–	*	*	R	–	*
AND eam, A	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) and (A)	–	–	–	–	–	*	*	R	–	*
OR A, #imm8	2	2	0	byte (A) ← (A) or imm8	–	–	–	–	–	*	*	R	–	–
OR A, ear	2	2	0	byte (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
OR A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
OR ear, A	2	3	0	byte (ear) ← (ear) or (A)	–	–	–	–	–	*	*	R	–	*
OR eam, A	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) or (A)	–	–	–	–	–	*	*	R	–	*
XOR A, #imm8	2	2	0	byte (A) ← (A) xor imm8	–	–	–	–	–	*	*	R	–	–
XOR A, ear	2	2	0	byte (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XOR A, eam	2 +	3 + (a)	(b)	byte (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–
XOR ear, A	2	3	0	byte (ear) ← (ear) xor (A)	–	–	–	–	–	*	*	R	–	*
XOR eam, A	2 +	3 + (a)	2 × (b)	byte (eam) ← (eam) xor (A)	–	–	–	–	–	*	*	R	–	*
NOT A	1	2	0	byte (A) ← not (A)	–	–	–	–	–	*	*	R	–	–
NOT ear	2	2	0	byte (ear) ← not (ear)	–	–	–	–	–	*	*	R	–	*
NOT eam	2 +	3 + (a)	2 × (b)	byte (eam) ← not (eam)	–	–	–	–	–	*	*	R	–	*
ANDW A	1	2	0	word (A) ← (AH) and (A)	–	–	–	–	–	*	*	R	–	–
ANDW A, #imm16	3	2	0	word (A) ← (A) and imm16	–	–	–	–	–	*	*	R	–	–
ANDW A, ear	2	2	0	word (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
ANDW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
ANDW ear, A	2	3	0	word (ear) ← (ear) and (A)	–	–	–	–	–	*	*	R	–	*
ANDW eam, A	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) and (A)	–	–	–	–	–	*	*	R	–	*
ORW A	1	2	0	word (A) ← (AH) or (A)	–	–	–	–	–	*	*	R	–	–
ORW A, #imm16	3	2	0	word (A) ← (A) or imm16	–	–	–	–	–	*	*	R	–	–
ORW A, ear	2	2	0	word (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
ORW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
ORW ear, A	2	3	0	word (ear) ← (ear) or (A)	–	–	–	–	–	*	*	R	–	*
ORW eam, A	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) or (A)	–	–	–	–	–	*	*	R	–	*
XORW A	1	2	0	word (A) ← (AH) xor (A)	–	–	–	–	–	*	*	R	–	–
XORW A, #imm16	3	2	0	word (A) ← (A) xor imm16	–	–	–	–	–	*	*	R	–	–
XORW A, ear	2	2	0	word (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XORW A, eam	2 +	3 + (a)	(c)	word (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–
XORW ear, A	2	3	0	word (ear) ← (ear) xor (A)	–	–	–	–	–	*	*	R	–	*
XORW eam, A	2 +	3 + (a)	2 × (c)	word (eam) ← (eam) xor (A)	–	–	–	–	–	*	*	R	–	*
NOTW A	1	2	0	word (A) ← not (A)	–	–	–	–	–	*	*	R	–	–
NOTW ear	2	3	0	word (ear) ← not (ear)	–	–	–	–	–	*	*	R	–	*
NOTW eam	2 +	3 + (a)	2 × (c)	word (eam) ← not (eam)	–	–	–	–	–	*	*	R	–	*

Note: For (a) to (c), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 15 Logic 2 (Long) [6 Instructions]

Mnemonic		#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ANDL	A, ear	2	5	0	long (A) ← (A) and (ear)	–	–	–	–	–	*	*	R	–	–
ANDL	A, eam	2 +	6 + (a)	(d)	long (A) ← (A) and (eam)	–	–	–	–	–	*	*	R	–	–
ORL	A, ear	2	5	0	long (A) ← (A) or (ear)	–	–	–	–	–	*	*	R	–	–
ORL	A, eam	2 +	6 + (a)	(d)	long (A) ← (A) or (eam)	–	–	–	–	–	*	*	R	–	–
XORL	A, ear	2	5	0	long (A) ← (A) xor (ear)	–	–	–	–	–	*	*	R	–	–
XORL	A, eam	2 +	6 + (a)	(d)	long (A) ← (A) xor (eam)	–	–	–	–	–	*	*	R	–	–

Note: For (a) and (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 16 Sign Reverse (Byte, Word) [6 Instructions]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
NEG	A	1	2	0	0	byte (A) ← 0 – (A)	X	–	–	–	–	*	*	*	*	–
NEG	ear	2	3	2	0	byte (ear) ← 0 – (ear)	–	–	–	–	–	*	*	*	*	–
NEG	eam	2 +	5 + (a)	0	2 × (b)	byte (eam) ← 0 – (eam)	–	–	–	–	–	*	*	*	*	*
NEGW	A	1	2	0	0	word (A) ← 0 – (A)	–	–	–	–	–	*	*	*	*	–
NEGW	ear	2	3	2	0	word (ear) ← 0 – (ear)	–	–	–	–	–	*	*	*	*	–
NEGW	eam	2 +	5 + (a)	0	2 × (c)	word (eam) ← 0 – (eam)	–	–	–	–	–	*	*	*	*	*

Note: For (a) and (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

Table 17 Absolute Values (Byte, Word, Long) [3 Instructions]

Mnemonic		#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
ABS	A	2	2	0	byte (A) ← Absolute value (A)	Z	–	–	–	–	*	*	*	–	–
ABSW	A	2	2	0	word (A) ← Absolute value (A)	–	–	–	–	–	*	*	*	–	–
ABSL	A	2	4	0	long (A) ← Absolute value (A)	–	–	–	–	–	*	*	*	–	–

Table 18 Normalize Instruction (Long) [1 Instruction]

Mnemonic		#	~	RG	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
NRML	A, R0	2	*1	1	0	long (A) ← Shift to where “1” is originally located byte (R0) ← Number of shifts in the operation	–	–	–	–	–	–	*	–	–	–

* : Set to 5 when the accumulator is all “0”, otherwise set to 5 + (R0).

Table 19 Shift Type Instruction (Byte, Word, Long) [27 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
RORC A	2	2	0	byte (A) ← With right-rotate carry	–	–	–	–	–	*	*	–	*	–
ROLC A	2	2	0	byte (A) ← With left-rotate carry	–	–	–	–	–	*	*	–	*	–
RORC ear	2	2	0	byte (ear) ← With right-rotate carry	–	–	–	–	–	*	*	–	*	*
RORC eam	2 +	3 + (a)	2 × (b)	byte (eam) ← With right-rotate carry	–	–	–	–	–	*	*	–	*	*
ROLC ear	2	2	0	byte (ear) ← With left-rotate carry	–	–	–	–	–	*	*	–	*	*
ROLC eam	2 +	3 + (a)	2 × (b)	byte (eam) ← With left-rotate carry	–	–	–	–	–	*	*	–	*	*
ASR A, R0	2	*1	0	byte (A) ← Arithmetic right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSR A, R0	2	*1	0	byte (A) ← Logical right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSL A, R0	2	*1	0	byte (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
ASR A, #imm8	3	*3	0	byte (A) ← Arithmetic right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSR A, #imm8	3	*3	0	byte (A) ← Logical right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSL A, #imm8	3	*3	0	byte (A) ← Logical left barrel shift (A, imm8)	–	–	–	–	–	*	*	–	*	–
ASRW A	1	2	0	word (A) ← Arithmetic right shift (A, 1 bit)	–	–	–	–	*	*	*	–	*	–
LSRW A/SHRW A	1	2	0	word (A) ← Logical right shift (A, 1 bit)	–	–	–	–	*	R	*	–	*	–
LSLW A/SHLW A	1	2	0	word (A) ← Logical left shift (A, 1 bit)	–	–	–	–	–	*	*	–	*	–
ASRW A, R0	2	*1	0	word (A) ← Arithmetic right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSRW A, R0	2	*1	0	word (A) ← Logical right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSLW A, R0	2	*1	0	word (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
ASRW A, #imm8	3	*3	0	word (A) ← Arithmetic right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSRW A, #imm8	3	*3	0	word (A) ← Logical right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSLW A, #imm8	3	*3	0	word (A) ← Logical left barrel shift (A, imm8)	–	–	–	–	–	*	*	–	*	–
ASRL A, R0	2	*2	0	long (A) ← Arithmetic right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSRL A, R0	2	*2	0	long (A) ← Logical right barrel shift (A, R0)	–	–	–	–	*	*	*	–	*	–
LSLL A, R0	2	*2	0	long (A) ← Logical left barrel shift (A, R0)	–	–	–	–	–	*	*	–	*	–
ASRL A, #imm8	3	*4	0	long (A) ← Arithmetic right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSRL A, #imm8	3	*4	0	long (A) ← Logical right barrel shift (A, imm8)	–	–	–	–	*	*	*	–	*	–
LSLL A, #imm8	3	*4	0	long (A) ← Logical left barrel shift (A, imm8)	–	–	–	–	–	*	*	–	*	–

Note: For (a) and (b), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: Set to 3 when R0 is 0, otherwise 3 + (R0).
*2: Set to 3 when R0 is 0, otherwise 4 + (R0).
*3: Set to 3 when imm8 is 0, otherwise 3 + imm8.
*4: Set to 3 when imm8 is 0, otherwise 4 + imm8.

Table 20 Branch 1 [31 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
BZ/BEQ rel	2	*1	0	Branch if (Z) = 1	–	–	–	–	–	–	–	–	–	–
BNZ/BNE rel	2	*1	0	Branch if (Z) = 0	–	–	–	–	–	–	–	–	–	–
BC/BLO rel	2	*1	0	Branch if (C) = 1	–	–	–	–	–	–	–	–	–	–
BNC/BHS rel	2	*1	0	Branch if (C) = 0	–	–	–	–	–	–	–	–	–	–
BN rel	2	*1	0	Branch if (N) = 1	–	–	–	–	–	–	–	–	–	–
BP rel	2	*1	0	Branch if (N) = 0	–	–	–	–	–	–	–	–	–	–
BV rel	2	*1	0	Branch if (V) = 1	–	–	–	–	–	–	–	–	–	–
BNV rel	2	*1	0	Branch if (V) = 0	–	–	–	–	–	–	–	–	–	–
BT rel	2	*1	0	Branch if (T) = 1	–	–	–	–	–	–	–	–	–	–
BNT rel	2	*1	0	Branch if (T) = 0	–	–	–	–	–	–	–	–	–	–
BLT rel	2	*1	0	Branch if (V) xor (N) = 1	–	–	–	–	–	–	–	–	–	–
BGE rel	2	*1	0	Branch if (V) xor (N) = 0	–	–	–	–	–	–	–	–	–	–
BLE rel	2	*1	0	Branch if ((V) xor (N)) or (Z) = 1	–	–	–	–	–	–	–	–	–	–
BGT rel	2	*1	0	Branch if ((V) xor (N)) or (Z) = 0	–	–	–	–	–	–	–	–	–	–
BLS rel	2	*1	0	Branch if (C) or (Z) = 1	–	–	–	–	–	–	–	–	–	–
BHI rel	2	*1	0	Branch if (C) or (Z) = 0	–	–	–	–	–	–	–	–	–	–
BRA rel	2	*1	0	Branch unconditionally	–	–	–	–	–	–	–	–	–	–
JMP @A	1	2	0	word (PC) ← (A)	–	–	–	–	–	–	–	–	–	–
JMP addr16	3	2	0	word (PC) ← addr16	–	–	–	–	–	–	–	–	–	–
JMP @ear	2	3	0	word (PC) ← (ear)	–	–	–	–	–	–	–	–	–	–
JMP @eam	2 +	4 + (a)	(c)	word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
JMPP @ear *3	2	3	0	word (PC) ← (ear), (PCB) ← (ear + 2)	–	–	–	–	–	–	–	–	–	–
JMPP @eam *3	2 +	4 + (a)	(d)	word (PC) ← (eam), (PCB) ← (eam + 2)	–	–	–	–	–	–	–	–	–	–
JMPP addr24	4	3	0	word (PC) ← ad24 0 – 15, (PCB) ← ad24 16 – 23	–	–	–	–	–	–	–	–	–	–
CALL @ear *4	2	4	(c)	word (PC) ← (ear)	–	–	–	–	–	–	–	–	–	–
CALL @eam *4	2 +	5 + (a)	2 × (c)	word (PC) ← (eam)	–	–	–	–	–	–	–	–	–	–
CALL addr16 *5	3	5	(c)	word (PC) ← addr16	–	–	–	–	–	–	–	–	–	–
CALLV #vct4 *5	1	5	2 × (c)	Vector call instruction	–	–	–	–	–	–	–	–	–	–
CALLP @ear *6	2	7	2 × (c)	word (PC) ← (ear) 0 – 15 (PCB) ← (ear) 16 – 23	–	–	–	–	–	–	–	–	–	–
CALLP @eam *6	2 +	8 + (a)	*2	word (PC) ← (eam) 0 – 15 (PCB) ← (eam) 16 – 23	–	–	–	–	–	–	–	–	–	–
CALLP addr24 *7	4	7	2 × (c)	word (PC) ← addr0 – 15, (PCB) ← addr16 – 23	–	–	–	–	–	–	–	–	–	–

Note: For (a), (c) and (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: Set to 3 when branch is executed, and 2 when branch is not executed.
*2: 3 × (c) + (b)
*3: Reads (word) of the branch destination address.
*4: W pushes to stack (word), and R reads (word) of the branch destination address.
*5: Pushes to stack (word).
*6: W pushes to stack (long), and R reads (long) of the branch destination address.
*7: Pushes to stack (long).

Table 21 Branch 2 [20 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
CBNE A, #imm8, rel	3	*1	0	Branch if byte (A) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CWBNE A, #imm16, rel	4	*1	0	Branch if word (A) ≠ imm16	–	–	–	–	–	*	*	*	*	–
CBNE ear, #imm8, rel	4	*1	0	Branch if byte (ear) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CBNE eam, #imm8, rel	4 +	*3	(b)	Branch if byte (eam) ≠ imm8	–	–	–	–	–	*	*	*	*	–
CWBNE ear, #imm16, rel	5	*1	0	Branch if word (ear) ≠ imm16	–	–	–	–	–	*	*	*	*	–
CWBNE eam, #imm16, rel	5 +	*3	(c)	Branch if word (eam) ≠ imm16	–	–	–	–	–	*	*	*	*	–
DBNZ ear, rel	3	*2	0	byte (ear) = (ear) – 1, Branch if (ear) ≠ 0	–	–	–	–	–	*	*	*	–	–
DBNZ eam, rel	3 +	*4	2 × (b)	byte (eam) = (eam) – 1, Branch if (eam) ≠ 0	–	–	–	–	–	*	*	*	–	*
DWBNZ ear, rel	3	*2	0	word (ear) = (ear) – 1, Branch if (ear) ≠ 0	–	–	–	–	–	*	*	*	–	–
DWBNZ eam, rel	3 +	*4	2 × (c)	word (eam) = (eam) – 1, Branch if (eam) ≠ 0	–	–	–	–	–	*	*	*	–	*
INT #vct8	2	14	8 × (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INT addr16	3	12	6 × (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INTP addr24	4	13	6 × (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
INT9	1	14	8 × (c)	Software interrupt	–	–	R	S	–	–	–	–	–	–
RETI	1	9	6 × (c)	Return from interrupt	–	–	*	*	*	*	*	*	*	–
RETIQ *6	2	11	*5	Return from interrupt	–	–	*	*	*	*	*	*	*	–
LINK #imm8	2	6	(c)	Stores old frame pointer in the beginning of the function, set new frame pointer, and reserves local pointer area	–	–	–	–	–	–	–	–	–	–
UNLINK	1	5	(c)	Restore old frame pointer from stack in the end of the function	–	–	–	–	–	–	–	–	–	–
RET *7	1	4	(c)	Return from subroutine	–	–	–	–	–	–	–	–	–	–
RETP *8	1	5	(d)	Return from subroutine	–	–	–	–	–	–	–	–	–	–

Note: For (a) to (d), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: Set to 4 when branch is executed, and 3 when branch is not executed.
*2: Set to 5 when branch is executed, and 4 when branch is not executed.
*3: Set to 5 + (a) when branch is executed, and 4 + (a) when branch is not executed.
*4: Set to 6 + (a) when branch is executed, and 5 + (a) when branch is not executed.
*5: Set to 3 × (b) + 2 × (c) when an interrupt request is issued, and 6 × (c) for return.
*6: This is a high-speed interrupt return instruction. In the instruction, an interrupt request is detected. When an interrupt occurs, stack operation is not performed, with this instruction branching to the interrupt vector.
*7: Return from stack (word).
*8: Return from stack (long).

Table 22 Miscellaneous Control Types (Byte, Word, Long) [36 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
PUSHW A	1	3	(c)	word (SP) ← (SP) – 2, ((SP)) ← (A)	–	–	–	–	–	–	–	–	–	–
PUSHW AH	1	3	(c)	word (SP) ← (SP) – 2, ((SP)) ← (AH)	–	–	–	–	–	–	–	–	–	–
PUSHW PS	1	3	(c)	word (SP) ← (SP) – 2, ((SP)) ← (PS)	–	–	–	–	–	–	–	–	–	–
PUSHW rlst	2	*3	*4	(PS) ← (PS) – 2n, ((SP)) ← (rlst)	–	–	–	–	–	–	–	–	–	–
POPW A	1	3	(c)	word (A) ← ((SP)), (SP) ← (SP) + 2	–	*	–	–	–	–	–	–	–	–
POPW AH	1	3	(c)	word (AH) ← ((SP)), (SP) ← (SP) + 2	–	–	–	–	–	–	–	–	–	–
POPW PS	1	3	(c)	word (PS) ← ((SP)), (SP) ← (SP) + 2	–	–	*	*	*	*	*	*	*	–
POPW rlst	2	*2	*4	(rlst) ← ((SP)), (SP) ← (SP) + 2n	–	–	–	–	–	–	–	–	–	–
JCTX @A	1	9	6 × (c)	Context switch instruction	–	–	*	*	*	*	*	*	*	–
AND CCR, #imm8	2	3	0	byte (CCR) ← (CCR) and imm8	–	–	*	*	*	*	*	*	*	–
OR CCR, #imm8	2	3	0	byte (CCR) ← (CCR) or imm8	–	–	*	*	*	*	*	*	*	–
MOV RP, #imm8	2	2	0	byte (RP) ← imm8	–	–	–	–	–	–	–	–	–	–
MOV ILM, #imm8	2	2	0	byte (ILM) ← imm8	–	–	–	–	–	–	–	–	–	–
MOVEA RWi, ear	2	3	0	word (RWi) ← ear	–	–	–	–	–	–	–	–	–	–
MOVEA RWi, eam	2 +	2 + (a)	0	word (RWi) ← eam	–	–	–	–	–	–	–	–	–	–
MOVEA A, ear	2	2	0	word(A) ← ear	–	*	–	–	–	–	–	–	–	–
MOVEA A, eam	2 +	1 + (a)	0	word (A) ← eam	–	*	–	–	–	–	–	–	–	–
ADDSP #imm8	2	3	0	word (SP) ← (SP) + ext (imm8)	–	–	–	–	–	–	–	–	–	–
ADDSP #imm16	3	3	0	word (SP) ← (SP) + imm16	–	–	–	–	–	–	–	–	–	–
MOV A, brgl	2	*1	0	byte (A) ← (brgl)	Z	*	–	–	–	*	*	–	–	–
MOV brg2, A	2	1	0	byte (brg2) ← (A)	–	–	–	–	–	*	*	–	–	–
MOV brg2, #imm8	3	2	0	byte (brg2) ← imm8	–	–	–	–	–	*	*	–	–	–
NOP	1	1	0	No operation	–	–	–	–	–	–	–	–	–	–
ADB	1	1	0	Prefix code for accessing AD space	–	–	–	–	–	–	–	–	–	–
DTB	1	1	0	Prefix code for accessing DT space	–	–	–	–	–	–	–	–	–	–
PCB	1	1	0	Prefix code for accessing PC space	–	–	–	–	–	–	–	–	–	–
SPB	1	1	0	Prefix code for accessing SP space	–	–	–	–	–	–	–	–	–	–
NCC	1	1	0	Prefix code for no change in flag	–	–	–	–	–	–	–	–	–	–
CMR	1	1	0	Prefix for common register bank	–	–	–	–	–	–	–	–	–	–
MOVW SPCU, #imm16	4	2	0	word (SPCU) ← (imm16)	–	–	–	–	–	–	–	–	–	–
MOVW SPCL, #imm16	4	2	0	word (SPCL) ← (imm16)	–	–	–	–	–	–	–	–	–	–
SETSPC	2	2	0	Enables stack check operation.	–	–	–	–	–	–	–	–	–	–
CLRSPC	2	2	0	Disables stack check operation.	–	–	–	–	–	–	–	–	–	–
BTSCN A	2	*5	0	Bit position of 1 in byte (A) from word (A)	Z	–	–	–	–	–	*	–	–	–
BTSCNS A	2	*6	0	Bit position (× 2) of 1 in byte (A) from word (A)	Z	–	–	–	–	–	*	–	–	–
BTSCND A	2	*7	0	Bit position (× 4) of 1 in byte (A) from word (A)	Z	–	–	–	–	–	*	–	–	–

Note: For (a) and (c), refer to “Table 4 Number of Execution Cycles in Addressing Modes” and “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: PCB, ADB, SSB, USB, and SPB : 1 state
DTB : 2 states
DPR : 3 states

*2: 3 + 4 × (number of POPs)

*3: 3 + 4 × (number of PUSHes)
*4: (Number of POPs) × (c), or (number of PUSHes) × (c)
*5: Set to 3 when AL is 0, 5 when AL is not 0.
*6: Set to 4 when AL is 0, 6 when AL is not 0.
*7: Set to 5 when AL is 0, 7 when AL is not 0.

Table 23 Bit Manipulation Instruction [21 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVB A, dir:bp	3	3	(b)	byte (A) ← (dir:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB A, addr16:bp	4	3	(b)	byte (A) ← (addr16:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB A, io:bp	3	3	(b)	byte (A) ← (io:bp) b	Z	*	–	–	–	*	*	–	–	–
MOVB dir:bp, A	3	4	2 × (b)	bit (dir:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB addr16:bp, A	4	4	2 × (b)	bit (addr16:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
MOVB io:bp, A	3	4	2 × (b)	bit (io:bp) b ← (A)	–	–	–	–	–	*	*	–	–	*
SETB dir:bp	3	4	2 × (b)	bit (dir:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB addr16:bp	4	4	2 × (b)	bit (addr16:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
SETB io:bp	3	4	2 × (b)	bit (io:bp) b ← 1	–	–	–	–	–	–	–	–	–	*
CLRB dir:bp	3	4	2 × (b)	bit (dir:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB addr16:bp	4	4	2 × (b)	bit (addr16:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
CLRB io:bp	3	4	2 × (b)	bit (io:bp) b ← 0	–	–	–	–	–	–	–	–	–	*
BBC dir:bp, rel	4	*1	(b)	Branch if (dir:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC addr16:bp, rel	5	*1	(b)	Branch if (addr16:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBC io:bp, rel	4	*1	(b)	Branch if (io:bp) b = 0	–	–	–	–	–	–	*	–	–	–
BBS dir:bp, rel	4	*1	(b)	Branch if (dir:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBS addr16:bp, rel	5	*1	(b)	Branch if (addr16:bp) b = 1	–	–	–	–	–	–	*	–	–	–
BBS io:bp, rel	4	*1	(b)	Branch if (io:bp) b = 1	–	–	–	–	–	–	*	–	–	–
SBBS addr16:bp, rel	5	*2	2 × (b)	Branch if (addr16:bp) b = 1, bit = 1	–	–	–	–	–	–	*	–	–	*
WBTS io:bp	3	*3	*4	Wait until (io:bp) b = 1	–	–	–	–	–	–	–	–	–	–
WBTC io:bp	3	*3	*4	Wait until (io:bp) b = 0	–	–	–	–	–	–	–	–	–	–

Note: For (b), refer to “Table 5 Correction Values for Number of Cycles for Calculating Actual Number of Cycles.”

*1: Set to 5 when branch is executed, and 4 when branch is not executed.
*2: 7 if conditions are met, 6 when conditions are not met.
*3: Indeterminate times
*4: Until conditions are met

Table 24 Accumulator Manipulation Instruction (Byte, Word) [6 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
SWAP	1	3	0	byte (A) 0 – 7 ↔ (A) 8 – 15	–	–	–	–	–	–	–	–	–	–
SWAPW/XCHW AL, AH	1	2	0	word (AH) ↔ (AL)	–	*	–	–	–	–	–	–	–	–
EXT	1	1	0	byte sign-extension	X	–	–	–	–	*	*	–	–	–
EXTW	1	2	0	word sign-extension	–	X	–	–	–	*	*	–	–	–
ZEXT	1	1	0	byte zero-extension	Z	–	–	–	–	R	*	–	–	–
ZEXTW	1	1	0	word zero-extension	–	Z	–	–	–	R	*	–	–	–

Table 25 String Instruction [10 Instructions]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVS/MOVSI	2	*2	*3	byte transfer @AH + ← @AL +, Counter = RW0	–	–	–	–	–	–	–	–	–	–
MOVSD	2	*2	*3	byte transfer @AH – ← @AL –, Counter = RW0	–	–	–	–	–	–	–	–	–	–
SCEQ/SCEQI	2	*1	*4	byte search (@AH +) – AL, Counter = RW0	–	–	–	–	–	*	*	*	*	–
SCEQD	2	*1	*4	byte search (@AH –) – AL, Counter = RW0	–	–	–	–	–	*	*	*	*	–
FISL/FILSI	2	5m + 6	*5	byte fill @AH + ← AL, Counter = RW0	–	–	–	–	–	*	*	–	–	–
MOVSW/MOVSWI	2	*2	*6	word transfer @AH + ← @AL +, Counter = RW0	–	–	–	–	–	–	–	–	–	–
MOVSWD	2	*2	*6	word transfer @AH – ← @AL –, Counter = RW0	–	–	–	–	–	–	–	–	–	–
SCWEQ/SCWEQI	2	*1	*7	word search (@AH +) – AL, Counter = RW0	–	–	–	–	–	*	*	*	*	–
SCWEQD	2	*1	*7	word search (@AH –) – AL, Counter = RW0	–	–	–	–	–	*	*	*	*	–
FILSW/FILSWI	2	5m + 6	*8	word fill @AH + ← AL, Counter = RW0	–	–	–	–	–	*	*	–	–	–

m: RW0 value (counter value)

*1: 3 when RW0 is 0, 2 + 6 × (RW0) when count out, and 6n + 4 when matched
*2: 4 when RW0 is 0, otherwise 2 + 6 × (RW0)
*3: (b) × (RW0)
*4: (b) × n
*5: (b) × (RW0)
*6: (c) × (RW0)
*7: (c) × n
*8: (c) × (RW0)

Table 26 Multiple Data Transfer Instructions [18 Instruction]

Mnemonic	#	~	B	Operation	LH	AH	I	S	T	N	Z	V	C	RMW
MOVM @A, @RLi, #imm8	3	*1	*3	Multiple data transfer byte ((A)) ← ((RLi))	–	–	–	–	–	–	–	–	–	–
MOVM @A, eam, #imm8	3 +	*2	*3	Multiple data transfer byte ((A)) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOVM addr16, @RLi, #imm8	5	*1	*3	Multiple data transfer byte (addr16) ← ((RLi))	–	–	–	–	–	–	–	–	–	–
MOVM addr16, @eam, #imm8	5 +	*2	*3	Multiple data transfer byte (addr16) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOVMW@A, @RLi, #imm8	3	*1	*4	Multiple data transfer word ((A)) ← ((RLi))	–	–	–	–	–	–	–	–	–	–
MOVMW@A, eam, #imm8	3 +	*2	*4	Multiple data transfer word ((A)) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOVMWaddr16, @RLi, #imm8	5	*1	*4	Multiple data transfer word (addr16) ← ((RLi))	–	–	–	–	–	–	–	–	–	–
MOVMWaddr16, @eam, #imm8	5 +	*2	*4	Multiple data transfer word (addr16) ← (eam)	–	–	–	–	–	–	–	–	–	–
MOVM @RLi, @A, #imm8	3	*1	*3	Multiple data transfer byte ((RLi)) ← ((A))	–	–	–	–	–	–	–	–	–	–
MOVM @eam, A, #imm8	3 +	*2	*3	Multiple data transfer byte (eam) ← ((A))	–	–	–	–	–	–	–	–	–	–
MOVM @RLi, addr16, #imm8	5	*1	*3	Multiple data transfer byte ((RLi)) ← (addr16)	–	–	–	–	–	–	–	–	–	–
MOVM @eam, addr16, #imm8	5 +	*2	*3	Multiple data transfer byte (eam) ← (addr16)	–	–	–	–	–	–	–	–	–	–
MOVMW@RLi, @A, #imm8	3	*1	*4	Multiple data transfer word ((RLi)) ← ((A))	–	–	–	–	–	–	–	–	–	–
MOVMW@eam, A, #imm8	3 +	*2	*4	Multiple data transfer word (eam) ← ((A))	–	–	–	–	–	–	–	–	–	–
MOVMW@RLi, addr16, #imm8	5	*1	*4	Multiple data transfer word ((RLi)) ← (addr16)	–	–	–	–	–	–	–	–	–	–
MOVMW@eam, addr16, #imm8	5 +	*2	*4	Multiple data transfer word (eam) ← (addr16)	–	–	–	–	–	–	–	–	–	–
MOVM bnk: addr16, bnk: addr16, #imm8*5	7	*1	*3	Multiple data transfer byte (bnk: addr16) ← (bnk: addr16)	–	–	–	–	–	–	–	–	–	–
MOVMWbnk: addr16, bnk: addr16, #imm8*5	7	*1	*4	Multiple data transfer word (bnk: addr16) ← (bnk: addr16)	–	–	–	–	–	–	–	–	–	–

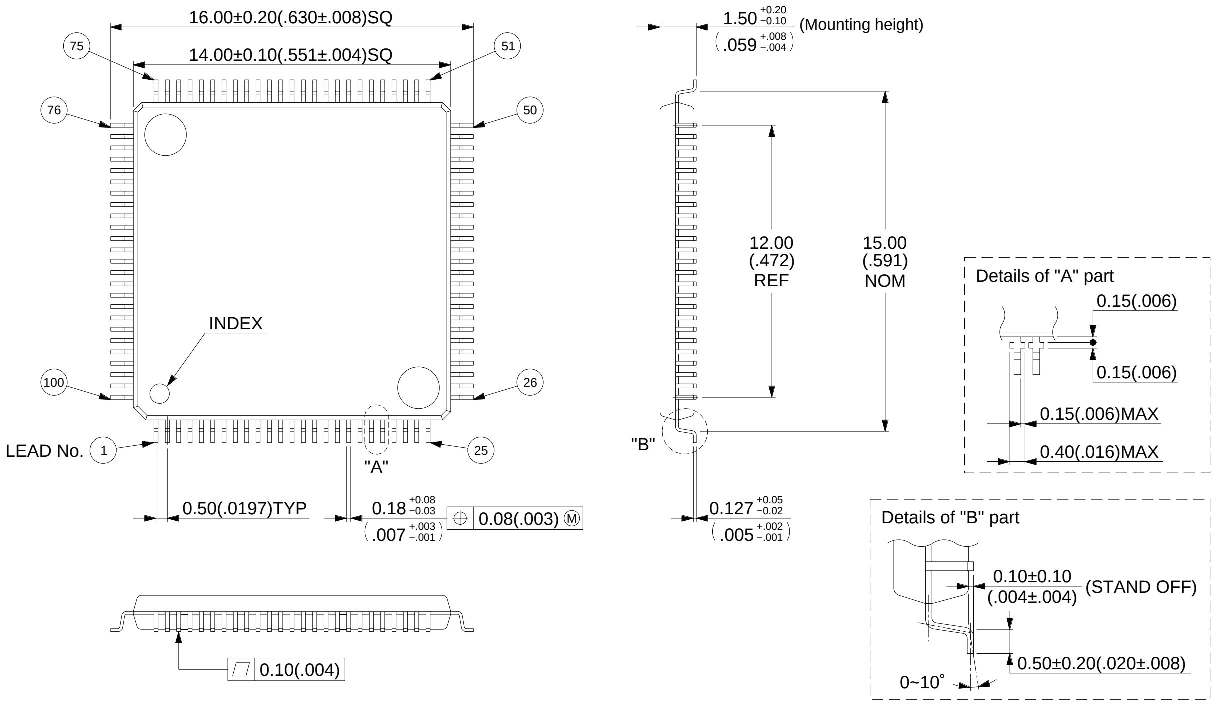
*1: 256 when 5 + imm8 × 5, imm8 is 0.
*2: 256 when 5 + imm8 × 5 + (a), imm8 is 0.
*3: (Number of transfer cycles) × (b) × 2
*4: (Number of transfer cycles) × (c) × 2
*5: The bank register specified by bnk is the same as that for the MOVS instruction.

■ ORDERING INFORMATION

Part number	Package	Remarks
MB90246APFV	100-pin Plastic LQFP (FPT-100P-M05)	

■ PACKAGE DIMENSIONS

100-pin Plastic LQFP
(FPT-100P-M05)



Dimensions in mm (inches)

FUJITSU LIMITED

For further information please contact:

Japan
FUJITSU LIMITED
Corporate Global Business Support Division
Electronic Devices
KAWASAKI PLANT, 4-1-1, Kamikodanaka
Nakahara-ku, Kawasaki-shi
Kanagawa 211-8588, Japan
Tel: 81(44) 754-3763
Fax: 81(44) 754-3329

http://www.fujitsu.co.jp/

North and South America
FUJITSU MICROELECTRONICS, INC.
Semiconductor Division
3545 North First Street
San Jose, CA 95134-1804, USA
Tel: (408) 922-9000
Fax: (408) 922-9179

Customer Response Center
Mon. - Fri.: 7 am - 5 pm (PST)
Tel: (800) 866-8608
Fax: (408) 922-9179

http://www.fujitsumicro.com/

Europe
FUJITSU MIKROELEKTRONIK GmbH
Am Siebenstein 6-10
D-63303 Dreieich-Buchschlag
Germany
Tel: (06103) 690-0
Fax: (06103) 690-122

http://www.fujitsu-ede.com/

Asia Pacific
FUJITSU MICROELECTRONICS ASIA PTE LTD
#05-08, 151 Lorong Chuan
New Tech Park
Singapore 556741
Tel: (65) 281-0770
Fax: (65) 281-0220

http://www.fmap.com.sg/

F98010